GUIDELINE-BASED DEVELOPMENT APPROACH FOR IOT HOME APPLICATIONS

NORIZAN BINTI A KAMALUDEEN

FACULTY COMPUTER SCIENCE & INFORMATION
TECHNOLOGY
UNIVERSITY OF MALAYA
KUALA LUMPUR

2019

Universiti Malaya

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IOT HOME APPLICATIONS

NORIZAN BINTI A KAMALUDEEN

DISSERTATION SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SOFTWARE ENGINEERING

FACULTY OF COMPUTER SCIENCE &
INFORMATION TECHNOLOGY
UNIVERSITY OF MALAYA
KUALA LUMPUR

2019

Universiti Malaya

University of Malaya

UNIVERSITY OF MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: Norizan Binti A Kamaludeen

Matric No: WGC 150009

Name of Degree: Master of Software Engineering

Title of Project Paper/Research Report/Dissertation/Thesis ("this Work"): Guideline-based development approach for IoT Home applications Field of Study:

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date:

Subscribed and solemnly declared before,

Witness's Signature Date:

Name:

Designation:

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IOT HOME APPLICATIONS

ABSTRACT

The concept of Smart Home is changing into a practice with more home devices and sensors are interconnected into an individual home infrastructure. The main objective of Smart Home is to make living safer, for security and peace of mind, convenience, comfort and well-being. Presently, developing IoT Home application systems progressively becomes crucial due to trending technologies and economic in the real world. IoT Home application development is different from any software system development as it additionally involves hardware development aspect to it. Most of the software developers do not have experience in hardware development. They are also lack of awareness of the significance of hardware development and its essential information as well as lack of skill in the development process. In this context, software developers and programmers are in the need of competent information with regards to hardware development or some directions in place for them to perform IoT application development. In the meantime, amid IoT development, hardware engineers familiar with IoT hardware may stand up to challenges with mobile or web application development due to inadequate experience or lack of information concerning it. Along these lines, hardware and software engineers clearly require assistance and guidance to work in tandem with IoT application development. Despite the fact that there have been some scattered guidelines important for IoT Home application development in different sources, they do not cover every one of the stages included, both in hardware and software development. Hence, the objective of this study is to propose a guideline-based development approach that provides comprehensive practices and instructions to guide software developers in IoT Home application development process more efficiently. The proposed approach comes in the forms of sequence of steps, checklists, best

Universiti Malaya

practices, and patterns. The proposed approach and the IoT Home application developed by employing these guidelines were evaluated by expert and user evaluation. The result of the evaluation shows that the proposed approach is efficient and valid. Meanwhile, the developed IoT Home application has also been evaluated to be effective. Moreover, the proposed guideline-based development approach aims to assist software developers throughout the development process starting from the requirements review process till the mobile application development process.

Keywords: IoT Home application, Guideline, Arduino, Smart Home.

Universiti Malaya

PENDEKATAN PEMBANGUNAN BERASASKAN GARIS PANDUAN UNTUK APLIKASI RUMAH IOT

ABSTRAK

Konsep Rumah Pintar berubah menjadi praktik dengan lebih banyak peranti rumah dan sensor saling berhubungan dengan infrastruktur rumah individu. Objektif utama Rumah Pintar adalah untuk menjadikan hidup lebih selamat, untuk keselamatan dan ketenangan fikiran, kemudahan, keselesaan dan kesejahteraan. Pada masa ini, membangunkan sistem aplikasi Rumah IoT secara progresif menjadi penting kerana teknologi dan ekonomi yang sedang berkembang di dunia nyata. Pembangunan aplikasi Home IoT adalah berbeza daripada sebarang pembangunan sistem perisian kerana ia juga melibatkan aspek pembangunan perkakasan kepadanya. Kebanyakan pemaju perisian tidak mempunyai pengalaman dalam pembangunan perkakasan. Mereka juga kurang kesedaran tentang pentingnya pembangunan perkakasan dan maklumat pentingnya serta kekurangan kemahiran dalam proses pembangunan. Dalam konteks ini, pemaju perisian dan pengaturcara sedang memerlukan maklumat yang kompeten berkaitan dengan pembangunan perkakasan atau beberapa arahan untuk mereka melaksanakan pembangunan aplikasi IoT. Sementara itu, di tengah pembangunan IoT, jurutera perkakasan yang biasa dengan perkakasan IoT mungkin menghadapi cabaran dengan pembangunan aplikasi mudah alih atau web kerana pengalaman yang tidak mencukupi atau kekurangan maklumat mengenainya. Sepanjang baris ini, jurutera perkakasan dan perisian jelas memerlukan bantuan dan bimbingan untuk bekerja seiring dengan pembangunan aplikasi IoT. Walaupun terdapat beberapa garis panduan tersebar yang penting untuk pembangunan aplikasi Home IoT dalam pelbagai sumber, mereka tidak meliputi setiap peringkat yang termasuk, dalam pembangunan perkakasan dan perisian. Oleh itu, objektif kajian ini adalah untuk mencadangkan pendekatan pembangunan berasaskan garis panduan yang menyediakan amalan dan arahan

Universiti Malaya

komprehensif untuk membimbing pemaju perisian dalam proses pembangunan aplikasi Rumah IoT dengan lebih cekap. Garis panduan yang dicadangkan ini datang dalam bentuk urutan langkah, senarai semak, amalan terbaik dan corak. Garis panduan yang dicadangkan dan aplikasi Rumah IoT yang dibangunkan dengan menggunakan garis panduan ini dinilai oleh pakar dan penilaian pengguna. Hasil penilaian menunjukkan bahawa panduan yang dicadangkan adalah cekap dan sah. Sementara itu, aplikasi Rumah IoT yang telah dibangunkan juga telah dinilai untuk menjadi berkesan. Selain itu, pendekatan pembangunan garis panduan yang dicadangkan bertujuan untuk membantu pemaju perisian sepanjang proses pembangunan bermula dari proses semakan keperluan sehingga proses pembangunan aplikasi mudah alih.

Kata kunci: Aplikasi Rumah IoT, Garis panduan, Arduino, Rumah Pintar.

Universiti Malaya

ACKNOWLEDGEMENTS

First of all, I express my deep gratitude to ALLAH, the Almighty and His Messager Muhammad S.A.W, for giving me the strength, peace of mind, endurance to overcome the obstacles and the ability to complete this dissertation. Thank you for showing me the directions to the right path and this dissertation becomes reality with HIS guidance.

I am very much obliged to my supervisor, Professor Dr. Lee Sai Peck for her direction, advice, untiring cooperation, enormous patience and motivation during the preparation of this dissertation. She showed me the correct way of doing right, and valuable research, and most importantly corrected the way of my thinking. I would like to express gratitude toward her for the diligent correction of my dissertation and constructive feedback throughout my research. I have favorable luck to work under her supervision.

My most heartfelt appreciation and gratitude go to my husband for his motivation and encouragement, and who are always by my side when I needed him most.

I would also like to extend a special note of thanks to my colleagues, friends, and people who have willingly helped me out with their abilities and provide valuable support and time towards this accomplishment.

Finally, I pray to Allah SWT to accept this dissertation as a sincere deed. Amin.

Universiti Malaya

TABLE OF CONTENTS

Universiti Malaya

Universiti Malaya

Universiti Malaya

Universiti Malaya

LIST OF FIGURES

Universiti Malaya

Universiti Malaya

Universiti Malaya

Universiti Malaya

LIST OF TABLES

Universiti Malaya

LIST OF APPENDICES

Universiti Malaya

CHAPTER 1: INTRODUCTION

1.1 Background

Internet of Things has been an active area of research since 2010. The “Internet of Things” (IoT) is a progressive topic of research around the world. The IoT can be defined as a system of interconnected computing devices, machine-like, and the digital device, objects, and animals or human that is provided with unique identifiers and the capability of data transmission in a network without involving human-to-human or human-to-computer interaction (Rouse, 2016).

The rapid growth of the IoT is changing our world and keeps on penetrating into almost every industry (Sundmaeker, et al. 2010). IoT domain related to Software Engineering can be divided into three main areas, which are embedded software that comes with smart things, analysis of data received from smart things, and applications that interact or communicate with the smart things. There are four main application classifications in the real-world result of the impact of IoT: (1) Personal and Home; (2) Enterprise; (3) Utilities; and (4) Mobile (Gubbi et al. 2013).

Meanwhile, IoT and Software Engineering are closely associated with each other as for the development of IoT software projects involved. This research targets to present software developers with an approach to improve or expedite the development process of IoT Home application. IoT Home application is specified as communication between everyday equipment and devices through the internet and controlled by other devices using a microcontroller in a home environment (Piyare & Lee, 2013).

As IoT explodes in the next few years, as positively expected, new challenges will emerge for software developers (Consel & Kabac, 2014). Security, privacy, complexity, efficiency, and competing standards will fuel the need for innovative IoT solutions, which could heavily impact by the choice of hardware and software tools, the

Universiti Malaya

programming language used and also the way software development for IoT are done (Alur et al. 2007). IoT software application development involves both hardware and software development. IoT software development is different from normal software development which involves hardware or smart devices, microcontroller, sensor, web application or mobile application (Patel et al. 2011). Currently, most software developers lack of hardware experience or are not familiar with the hardware which involves electric and electronic devices. IoT software application development consists of the hardware which is the smart thing or smart devices that will communicate with each other and are controlled by software applications such as mobile applications. In this situation, the software developer or programmer is in need of sufficient knowledge on the hardware or some guidance in order for them to do IoT software application development (Kortuem et al. 2013). Despite that, computer engineers familiar with the IoT hardware might face difficulties with the software application development process as they do not have the knowledge or enough experience regarding it. As such, both hardware and software engineers obviously require assistance and guidance in developing IoT software application development.

This research aims to propose a guideline-based development approach that could help the developers on the IoT Home application development and ease the development process. The guideline-based development approach basically provides software engineers with guidelines or steps of each development stage in a proper way, with the purpose of building better software systems (Ramachandran, 2012; Hneif & Ow, 2009).

Universiti Malaya

1.2 Problem Statement

There are many guidelines available in different sources including websites, journal and conference papers, books, and technical reports from software companies related to IoT Home application. Mostly, each source focuses on certain parts of software development or hardware development, for instances, some websites focus on guidelines for sensor communication to smart devices or only on the Android mobile application development; some articles focus on the guidelines for deployment of the IoT software application only (Oriwoh et al. 2013). The guidelines that are presented in different sources are however not all reliable. Hence, manually looking and collecting suitable guidelines to develop IoT Home application development is not efficient, since it may take extra time and effort. Other than that, software developers or programmers might overlook some crucial guidelines that could cause some difficulty in the development.

IoT Home application development consists of hardware development, software development, the interaction between smart devices, internet communication and analysis of data. The hardware development related to smart devices is a great challenge for fresh software developers as well as for developers who are not acquainted with hardware. Thus, the steps or guidelines on setting up the hardware or smart devices as well as the programming of the hardware will likely be a great help for the developers who are not familiar with the setting up of IoT hardware for the IoT development, and it will reduce dependability to hardware developers (i.e., electric and electronic engineers) concerning this process. Other than that, the guidelines in the hardware setup, as well as the microcontroller board programming (e.g., Arduino board programming) will help the developers to speed up the development as the developers have the least knowledge of the hardware. As mentioned above, without a proper set of guidelines on setting up the hardware for the IoT Home application development such

Universiti Malaya

as the sensors, the board to connect multiple sensors, and board programming, the developers would have a hard time in IoT Home application development as they need to depend on the help from people with different areas of expertise such as network or electronic background.

The choice of language is also really important to decide at the early stage of the design of any IoT application development. Choosing an unsuitable programming language for development will cost time and resources as well as the failure of any project. This research will help the user to select the best programming language that will be used for the IoT Home application development based on multiple factors including: 1) the operating system, 2) the programming knowledge, 3) hardware and etc. The speed of development will not be affected if the developers manage to find the most suitable programming language without much time spent and if they know that the language will not be a problem or a factor that will affect the quality attributes of the system. There is lack of guideline on choosing the most suitable programming language. Language selection is very important as it will impact the system's dependability, security part, and system's performance, and also the adaptation of the software when there is a change of requirement on the IoT Home application development.

As of now, as far as the researcher knows, there is lack of guidelines or a comparative study on the guidelines to help developers in aspects includes requirement analysis and design specific, hardware-specific, and programming-specific of IoT Home application development. Guidelines provide information and knowledge resulting from several years of successful best practice and experience in previous projects (Ramachandran, 2012). This saves time, money and effort with the quality we all want and guidelines shape principles from theories, laws and observations (Ramachandran, 2012).

Universiti Malaya

1.3 Motivation

Reflecting back to the problem statement, the lack of suitable guidelines and approach on the IoT Home application development fuels this research motivation.

This study aims to help the developers by coming up with a guideline-based development approach from the early stage till the end of the development stage covers: 1) requirement review, 2) development environment preparation, 3) hardware setup, 4) hardware programming, and 5) mobile application programming for the IoT Home application development. Apart from that, to present guidelines to assist the developers in choosing the right programming language based on the operating system, Android or iOS for IoT Home application development. Also, this study includes proposing a guideline in the coding of the hardware board programming and the setting up of the hardware part which consists of devices, sensors, and the board for IoT Home application development.

The outcome of this study will serve as a consulting basis for programmers or developers in the area of the IoT Home application development and also provide a systematic exploration of existing work and suggestion of a number of potentially important guidelines and directions.

Universiti Malaya

1.4 Research Objectives

This study aims to help software developers and stakeholders in the development of IoT Home application development. The objectives of this research are:

- To identify the problem and challenges faced by the software developers on the IoT Home application development process.
- To identify the available guidelines for the IoT home application development.

- To propose guideline-based development approach for developers includes the requirement review stage till the mobile application stage in order to improve the IoT Home application software development process.
- To develop a prototype for the IoT Home application on controlling the Home appliances by applying the proposed guideline-based development approach.
- To validate the proposed guideline-based development approach and the prototype based on the user evaluation and expert review.

1.5 Research Questions

The research questions defined in this research are:

- What are the problem and challenges faced by the software developers on the IoT Home application development process?
- Are there any guidelines exist from the early stage till the end of the development stage for IoT Home application development?
- What is the guideline-based development approach for software developers that could help to ease the IoT Home application development process?
- What kind of prototype to be developed by applying the proposed guideline-based development approach?
- How to evaluate the proposed guideline-based development approach and the developed prototype?

1.6 Research Scope

This research focuses on IoT application development guidelines. In this study, the research is narrowed to IoT Home application development guideline approach.

Universiti Malaya

The main focus of this dissertation is:

1. Focus on IoT Home application development
2. Propose a guideline-based development approach for software developers
3. Develop a prototype on the IoT Home mobile application development based on the proposed guideline-based development approach.

The main purpose of this dissertation is to help and guide the user and the software developers involved in the IoT Home application development. The proposed guideline-based development approach can be used as a basic guide for both newcomers and professionals who working the IoT Home mobile application development. The prototype will be evaluated based on the usability as well as expert review.

Universiti Malaya

1.7 Research Approach

This section explains the research approach in accomplishing the research objectives of this study. There are five stages to be taken to complete the purpose of this project: 1) Identification of the problem stage; 2) direction or analysis for solution stage; 3) proposing guideline-based development approach; 4) system design, and development stage; and 5) evaluation stage as depicted in Figure 1.1.

The research approach is further elaborated in Chapter 3.

The first stage is the identification of the problem stage. In this stage, a background study on the Internet of Things (IoT), IoT related to software development and software engineering is conducted. Previous work and the related work on the IoT and software engineering as well as software development domain are also studied. Initial research objectives and research questions related to this research are identified.

The second stage focuses on direction or analysis for the solution stage where a literature review is carried out. At this stage, the existing research on the IoT and software development guideline-based development approach is studied. The literature review is further narrowed down to IoT Home application development. Identification of the solution related to the guideline-based development approach for IoT Home application development is defined. Furthermore, the research objectives and the research question identified in first stage are redefined.

Proposing guideline-based development approach forms the third stage of this research. The guideline-based development approach for IoT Home application development is proposed at this stage. The guidelines are presented in the following categories: patterns, checklist, sequence of steps and best practices.

System design and development forms the fourth stage of this research. In this stage, an IoT Home application is developed based on the approach proposed at the third stage. The system is designed and the approach is referred during the development.

The Final stage is on evaluation. In this stage, the developed IoT Home application is evaluated by users and expert review. The evaluation is done based on the usability. The guideline-based development approach is also evaluated by experts and users. The outcome of the evaluation is documented and analyzed.

Universiti Malaya

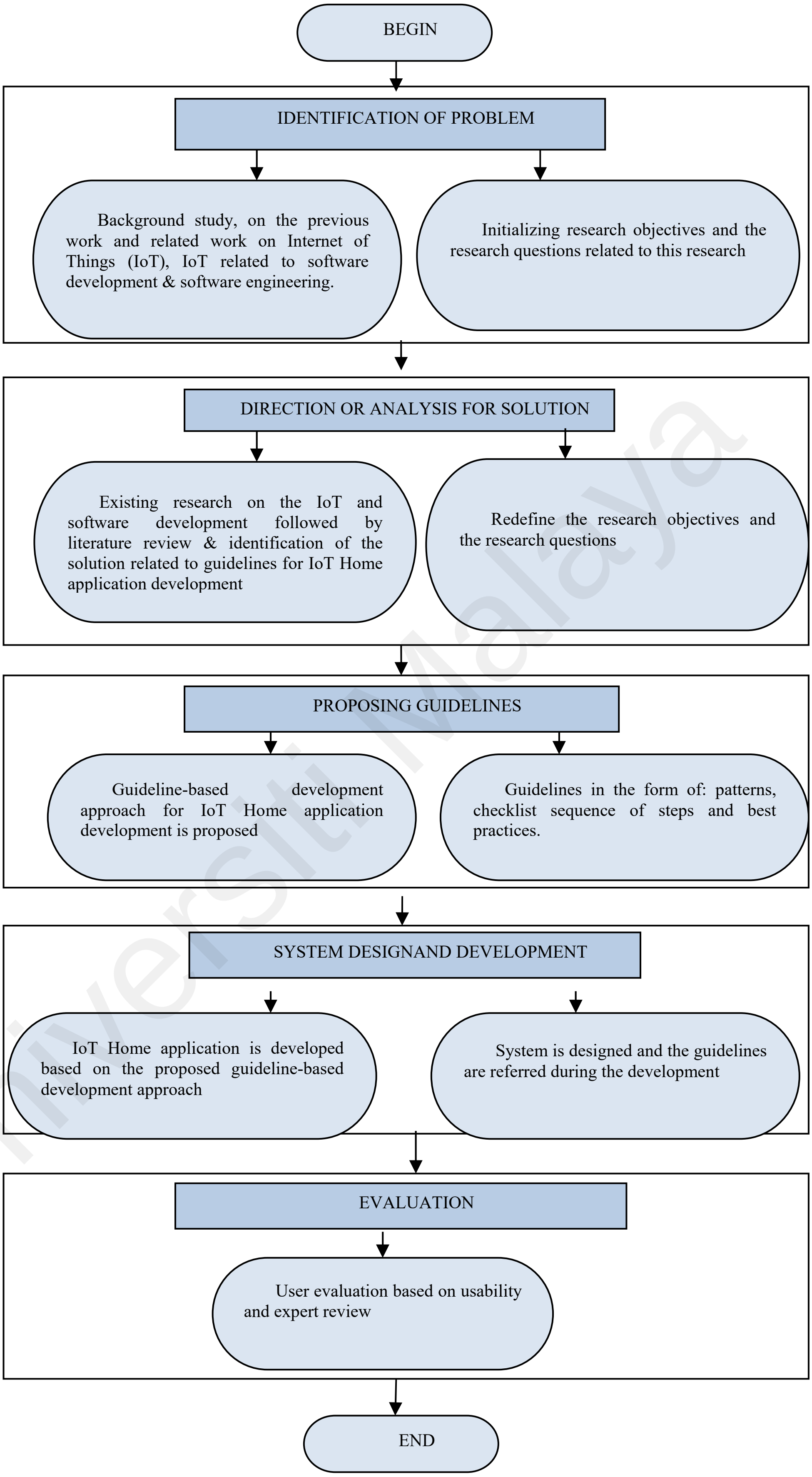

BEGIN
IDENTIFICATION OF PROBLEM
Background study, on the previous work and related work on Internet of Things (IoT), IoT related to software development & software engineering.
Initializing research objectives and the research questions related to this research
DIRECTION OR ANALYSIS FOR SOLUTION
Existing research on the IoT and software development followed by literature review & identification of the solution related to guidelines for IoT Home application development
Redefine the research objectives and the research questions
PROPOSING GUIDELINES
Guideline-based development approach for IoT Home application development is proposed
Guidelines in the form of: patterns, checklist sequence of steps and best practices.
SYSTEM DESIGNAND DEVELOPMENT
IoT Home application is developed based on the proposed guideline-based development approach
System is designed and the guidelines are referred during the development
EVALUATION
User evaluation based on usability and expert review
END

Figure 1.1 : Research Methodology

Universiti Malaya

1.8 Significance of Research

IoT application development is becoming a very popular project in today's world. Software developers have started developing all kinds of development related to IoT especially home applications. There are a lot of guidelines exist but scattered all over places in the websites, journals, books, and etc. A software developer may fail to notice essential and valuable guidelines while developing because of inadequate search capabilities, restricted access to sources of guidelines and challenges done manipulating a large number of guidelines. Therefore, a complete guideline-based development approach from the early till the end of the development process proposed in this research could be very helpful, in saving the time on searching for guidelines from different sources. In addition, the speed of the development will be taken care of, as most of the required information for the IoT Home application development will be made available in this study.

The significance of this research is to guide the software developers on setting up the hardware part without depending on an electronic engineer expert for setting hardware equipment as the developer might be lack of experience in this matter. This guideline-based development approach will be very helpful for inexperienced developers. Besides that, this research will guide the developers in deciding a suitable programming language based on few criteria for Android and iOS mobile applications for IoT Home application development. Junior developers or fresh graduates in this area of development will find this guideline-based development approach very valuable as it will guide them in software requirement analysis that is the first stage of development, the preparation of the hardware and software development environment till the last stage of the development.

Universiti Malaya

Overall, the study is to help and guide developers in developing IoT Home mobile application development without any dependencies on other experts or having to search various resources for guidelines.

1.9 Dissertation Organization

This dissertation is organized into seven chapters. Chapter 2 presents a literature review on IoT and its related development, IoT Home application development, and guidelines. Chapter 3 discusses the research methodology in more detail. Chapter 4 presents the proposed approach on the IoT Home application development. There are six proposed guidelines from the early stage of the development till the completion. Chapter 5 presents the development of the IoT Home application which implements the proposed guideline-based development approach throughout all IoT Home application development phases, from requirements review until evaluation the IoT Home application. Chapter 6 describes the testing and evaluation of IoT Home application development to ensure that the objectives are fulfilled. Finally, Chapter 7 discusses the contribution of the research and some suggestion for future work.

Universiti Malaya

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

IoT is extensively enforced in the smart home, healthcare, government, security, industrial, transportation, and etc. IoT technology creates relations or communication between all smart things or objects and the internet through sensor devices such as Radio Frequency Identification (RFID), and IR sensors which involve sending or receiving of data and controlling smart devices (Li & Yu, 2011).With the increase of the economic trend and technologies used in the current world, building IoT home application systems gradually is becoming more important. It is very important to build or develop an IoT home application system by applying software development methods to ensure the developed system fulfill the requirement, and quality wise, assist in organizing and speeding up the development process. To develop an IoT home application system that consists of many phases, software engineers need to choose a development approach which provides guidelines for every stage of the development process to assist the developers till the development is completed.

A software development method provides guidance during the software development process of a software system. Software engineers apply a software development method to assist them throughout the development process. There are various types of software development methods supporting various approaches such as Agile development approach, Waterfall development, Object Oriented approach, Guideline-based development approach, and many more (Farrell, 2007; Ramachandran, 2012). The guideline-based approach is defined as a software development method that presents software developers with a set of guidelines or information for performing every stage of the development in a more effective approach (Ramachandran, 2012; Malik & Hneif, 2010).

Universiti Malaya

The following sections of this chapter presents related work on IoT, offers related work on IoT Home application development and its features, presents an overview of guideline-based development approach, and concludes the literature review chapter.

2.2 Internet of Things

IoT, an innovation that conveys life to the smart objects such as home appliances, cars, building security, printers, and vending machines, with the intention to collect information from the devices and communicate with the smart devices (Atzori et al. 2010).

Various research works have been proposed for IoT development involving such as frameworks, programming model, programming techniques and guideline approach. In this section, the IoT application areas, tools, standards, technologies, characteristics, and various works involving IoT are discussed.

2.2.1 The Characteristic of IoT Software Projects

The characteristics of IoT software application has been highlighted by Patel et al (2016). The different characteristics identified are: (1) heterogeneous components - various types of components, (2) heterogeneous platforms – operating platforms, (3) heterogeneous runtime system – in different runtime systems, (4) heterogeneous interaction modes – various terms of how information or data can be retrieved, (5) static topology or dynamic topology, (6) the network size – quantity of devices partaking in an application (Patel et al, 2016). Moreover, Miorandi et al (2012) list the IoT features that the development should take into count: things heterogeneity, scalability, ubiquitous, energy-optimized solutions, localization and tracking ability, self-

Universiti Malaya

organization ability, semantic interoperability and data management, and embedded security and privacy-preserving mechanisms.

The IoT software development is built grounded on the three criteria interconnected to the capability of the things to: (i) identification (anything that is identifiable), (ii) communication (anything that can communicate), and (iii) interaction (anything that can interact) (Miorandi et al. 2012). Furthermore, the authors explain the things as a physical embodiment and a set of related physical features, communication functionalities at a minimum level, has a unique identifier and computing abilities, and is able to interact with other physical things.

2.2.2 IoT Development and Programming

Programming is one of the important tasks in the IoT development process. In (Namiot & Sneps-Sneppe, 2014), the authors write on the challenges faced with regards to the IoT programming, which are, power supply, latency, network topology and finding suitable tools for managing the data from the devices. The authors discuss the programming models: (1) reactive programming, (2) Abstract Task Graph (ATaG), (3) Computational REST, (4) Flow-Based Programming (FBR), and (5) Actor Model, and the current and future system software platforms for Machine to Machine communications (M2M) (Namiot & Sneps-Sneppe, 2014**).** Lu et al. (2015) recognizes dependability and programmability improvement are the challenges. For the programmability, the authors state that there is a need to make software development easier for the end-user or the stakeholder to involve in the development and customization. They have suggested for developing a high-level modeling language that can be auto-compilated towards executable code. They have concluded the need to

Universiti Malaya

increase the level of automation in development at the level of abstraction and modularity.

The IoT software development is built grounded on the three criteria interconnected to the capability of the things to: (i) identification (anything that is identifiable), (ii) communication (anything that can communicate), and (iii) interaction (anything that can interact) (Miorandi et al. 2012). Furthermore, the authors explain the things as a physical embodiment and a set of related physical features, communication functionalities at a minimum level, has a unique identifier and computing abilities, and is able to interact with other physical things.

System and application software are discussed in (Mzahm et al. 2015), which the authors further discuss on the programming software which is divided into three different stages, which are machine language, assembly language and high-level language. The different frameworks which support different programming languages that every developer can select to use during software design are discussed. In this research, the authors have introduced a software spectrum to define the best software requirements for the hardware Agent of Things applications.

Furthermore, the authors in (Díaz et al. 2009) have separated IoT application development process into several concerns. The concerns involve domain, functional, deployment, platform, the connection part and the evolution of the things. In (Consel & Kabac, 2014), the main research question in their study was in which language should be selected for the development of the Internet of Things software. The authors mentioned that language selection is very important as it would impact the system's dependability, security, and system's performance, and also the adaption of the software when there is a change of requirement. Furthermore, the authors have compared and

Universiti Malaya

discussed the C language and the Ada language. In this article, the strong point and weak point as well as significance of the use of both languages are discussed.

The survey in (IoT Developer Survey. 2016) reveals that the top 6 programming languages used in the development of IoT related development are Java, C, JavaScript, Python, Node.js, and C++. This survey indicates that the current technologies focused by developers are more on embedded development, web development and Big Data for the IoT. This establishes that embedded development and web development as top concerns of the IoT focused development technology. The most frequently used IoT operating system is Linux for the IoT devices.

Universiti Malaya

2.2.3 IoT and Tools

IoT tools are for software developers to build IoT systems or applications. IoT tools are considered a solution used by software developers in developing IoT software systems. DiaSuite is a tool-based methodology introduced by (Consel & Kabac, 2014) to guide developers in developing IoT applications. The methodology depends on a compiler that produces support of the Java programming framework. The authors mention that their programming framework helps developers guarantee conventionality among the design and implementation phased. Their approach provides developers with a back-end to address the deployment and implementation of the applications. As such, DiaSuite is a tool that is developed to focus on the programming or development level and it is a design-driven methodology (see Table 2.1).

In (Mazzei et al. 2015), Mazzei et al. focus on programming for embedded development in the IoT domain. The authors introduced a design suite called Viper Suite for IoT prototyping. The design suite is defined and a likely execution of a software suite for the design of smart devices is presented. The article offers the progress of the Viper suite, which is focused on the programming of embedded devices of the IoT domain. The Viper suite is developed using Python Embedded in real time and it is an open-source development suite adapted to IoT. The result is for the use of IoT inventors, software developers, programmers as well as application providers (see Table 2.1).

2.2.4 IoT and Frameworks

The embedded systems regularly developed via low-level languages such as C language, on top of the operating system, which allows to control the accessible resources and to make changes for improvement. Even so, using C language for development has usually made testing difficult and the system hard to maintain (Sivieri et al. 2012). Due to that, ELIOT, an Erlang-based development framework for an IoT embedded development has been presented in this article (see Table 2.1).

Raphael Hiesgen (Hiesgen et al. 2015) introduced the C++ Actor Framework that facilitates for IoT development using C++ language. In this study, Erlang and Scala programming is discussed as actor based programming languages. The authors concentrated on development using the actor model. They mention that using the actor model in the development would concentrate on the application logic rather than focusing on the low-level coding. They presented this approach to raise the quality and the strength of code, and reduce the code dependencies based on hardware or systems, as this is immediately required in the IoT domain. As such, it is known that the authors have focused on the challenges at the programming stage for the embedded development (see Table 2.1).

Universiti Malaya

An IoT application development Framework called IoTSuite is introduced by (Patel et al. 2016). In addition, the authors also guide the programmers in picking a method by comparing their approach to other IoT application development that exists from different extents. The IoTSuite is successfully applied in designing, implementing and deploying IoT smart home application (Patel et al. 2016) (see table 2.1).

Table 2.1: Comparison study of existing IoT Tools and Frameworks focused on IoT software development

Tools / Frameworks	Programming language	Method	Target user	Limitation	Mobile/Web-based
Test	Java	Tool-based methodology	Programmers	Focus only at the programming level. Lack of guidelines provided on the IoT hardware set up	Mobile and Web-based
Viper Suite	Python	Open source development suite	Developers, IoT designers, programmers	Focus only at the programming level. Lack of guidelines provided on IoT hardware set up and IoT Mobile application development.	Mobile

Universiti Malaya

Table 2.1 continued

Tools / Frameworks	Programming language	Method	Target user	Limitation	Mobile/Web-based
ELIOT	Erlang	Erlang-based development framework	Developers	Focus only at the programming level. Lack of guidelines provided on IoT hardware set up and IoT Mobile application development.	
C++ Actor Framework	C++ language	Framework	Developers	Lack of guidelines provided to develop IoT applications from early stage till the deployment stage. Focus on programming only.	
IoTSuite	Java	Framework	Developers	Lack of guidelines provided on IoT hardware set up.	Mobile and Web-based

Universiti Malaya

2.2.5 Applications associated with IoT

In the IOT software development, three main components are identified, that is (a) hardware (e.g. actuators, sensor), (b) middleware, and (c) presentation (Gubbi et al. 2013). Due to the IoT outcome, there are likely developments on the enormous number of applications. In (Gubbi et al. 2013), IoT applications are grouped in four categories and each category has its application area, 1) Home - security, entertainment, health, utilities and appliances, 2) Transport - logistics, traffic, parking, emergency services, and highways, 3) Community - factory, retail, surveillance, smart meeting and business intelligence, and 4) National - infrastructure, defense, remote monitoring, and utilities. On the other hand, the authors in (Atzori et al. 2010) have classified IoT applications into four categories which are personal and social, healthcare, transportation and logistics, smart environment (office, plant, home), and advanced applications. Additionally, the authors elaborate the potential applications on each category. The applications are, for personal and social (social networking historical queries, losses and theft), for healthcare (tracking, identification, authentication, data collection, sensing), for transportation and logistics (logistics, assisted driving, mobile ticketing, environment monitoring, augmented maps), for smart environment (comfortable home/offices, industrial plants, smart museum and gym), and for the advanced applications (robot taxi, city information model, enhanced game room).

Gubbi et al. (2013) highlights in Figure 2.1 the related IoT application areas and the categories.

Universiti Malaya

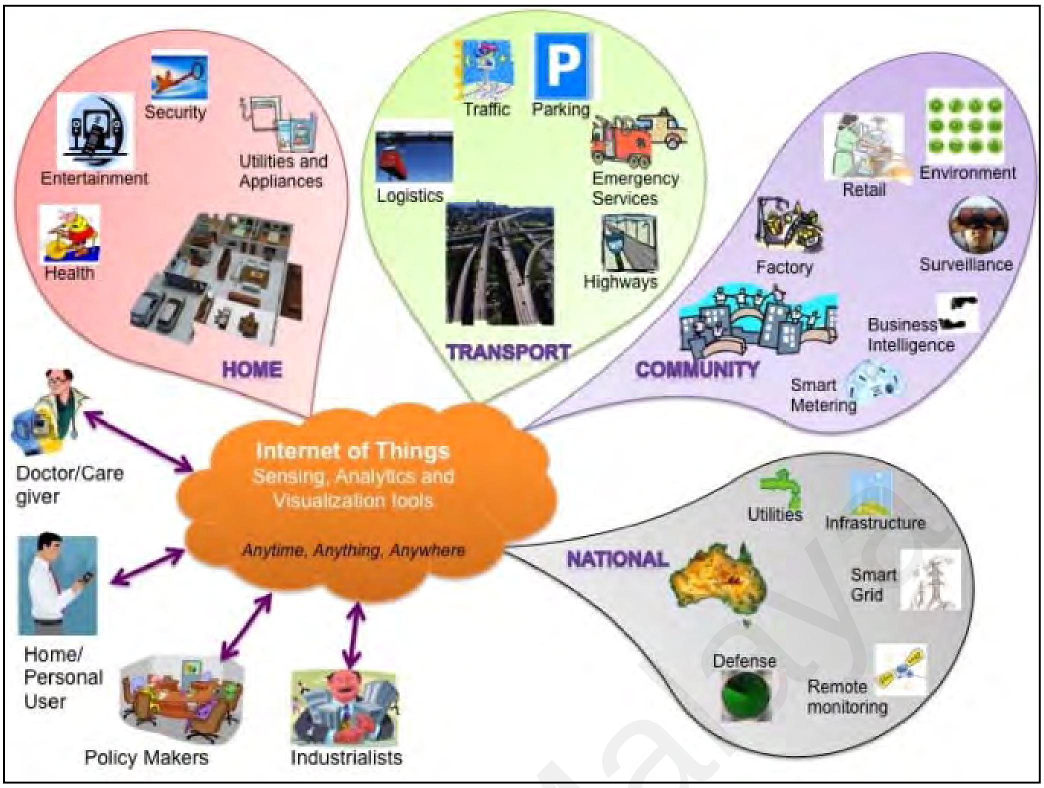


Figure 2.1 : IoT application areas and categories (Gubbi et al. 2013)

In recent times, the AGILE IoT, Eclipse IoT Working Group, and IEEE IoT together have conducted an IoT developer survey (IoT Developer Survey. 2016) which shows that the top five application or area concerns are home automation, IoT platforms, Industrial automation, energy management and connected cities. Xu et al. (2014) discuss industrial IoT applications: health care service, workplace and home support, accounting and production management, ecological monitoring, food supply chain, transportation, security, and surveillance. The work in (Kess & Kropsu-Vehkapera, 2016) describes that IoT can be useful in almost every part of individual and business life which are buildings, IT and networks, security and safety, retail, transportation, industrial, life science and health, consumer and home, energy, and buildings.

Universiti Malaya

2.2.6 Technologies associated with IoT

The architecture that is commonly used in the IoT is Service Oriented Architecture (SOA) approach. The SOA based architecture layers from bottom up are: (1) Objects, (2) Object abstraction, (3) Service Management, (4) Service composition, and (5) Applications layer (Atzori et al. 2010). Xu et al. (2014) describe the SOA in four layers consisting of Sensing layer (hardware) (e.g. RFID), Networking layer, Service layer and an Interface layer (interaction part). In the design part of the IoT application, the authors have mentioned that the energy, latency, throughput, scalability, topology, security and safety are considered.

Currently, five IoT technologies that is important in the IoT focus devices or services deployment are highlighted in (Lee & Lee, 2015). The technologies are: (1) Radio Frequency Identification (RFID), (2) Wireless Sensor Networks (WSN), (3) Middleware, (4) Cloud Computing, and (5) IoT application software. The authors in (Lee & Lee, 2015) further explain each of these technologies. RFID and WSN are the most common and well-known technologies used for IoT embedded development (Lu et al. 2015; Mzahm et al. 2015; Díaz et al. 2009; Consel & Kabac, 2014). Those technologies are already used in the IoT based products. The authors further elaborate on the usage and the connection of the technologies with the IoT things and services. Additionally, the authors in (Lee & Lee, 2015) describe the future technologies that would be in the market from 2015 to 2020 and beyond 2020 are: the network (e.g. network context awareness), software (e.g. distributed intelligence, things-to-things collaboration environments), hardware (e.g. smart sensors) and data processing (e.g. context-aware data processing and data responses).

In the IoT software development, three main components are identified that is: (a) hardware (e.g. actuators, sensor), (b) middleware, and (c) presentation (Gubbi et al.

Universiti Malaya

2013). The authors in (Gubbi et al. 2013) further elaborated the three components in detail. The actuators, the sensor will be Radio Frequency Identification (RFID), Wireless Sensor Networks (WSN), WSN hardware, and WSN communication stack. The middleware is a method to merge internet communications with Service Oriented Architecture (SOA) and sensor connection to give access to heterogeneous sensor resources in a deployment. Meanwhile, the presentation is defined as a novel straightforward representation and understanding tools which can be generally accessed on various stages and which can be designed for various applications.

2.2.7 IoT related Open Sources and Standards

The Eclipse IoT (Eclipse IoT Website, 2015) has the open source for IoT software development. In (Eclipse IoT Website, 2015), there are open source available for the IoT devices based on the following criteria: hardware - Eclipse Edje (which is to enable connection to the hardware elements from the IoT devices), communication- Eclipse Paho (by using MQTT protocol, it makes possible communication for IoT devices), and remote management: Eclipse Wakaama (remotely control of IoT devices). Eclipse IoT also provides open source for IoT gateway which are Eclipse Kura, Eclipse Smart Home, and Eclipse 4diac. Eclipse Kura is a framework that provides a useful middleware and application container for the IoT gateway. The application container for Eclipse Kura is Eclipse Equinox or Eclipse Concierge. Eclipse Smart Home provides a platform,particularly for the home automation. Meanwhile, Eclipse 4diac is a framework for the industrial automation. Eclipse IoT (Eclipse IoT Website, 2015) also provides the open source for the IoT cloud platforms: Eclipse Kapua, Eclipse OM2M, Eclipse Hono, Eclipse Mosquitto, Eclipse Leshan, Eclipse hawkBit, and Eclipse BIRT.

Universiti Malaya

Eclipse IoT is supporting open standards and provides open source usage to IoT protocols and standards, for example, CoAP (Constrained Application Protocol) , Datagram Transport Layer Security (DTLS), PPMP (Production Performance Management Protocol), MQTT, international standard ISO/IEC 15118, IEC 61499, industry standard OMA Lightweight M2M (LWM2M),Open Geospatial Consortium (OGC) standard, OPC Unified Architecture (UA) and one M2M. The authors in (Kess & Kropsu-Vehkapera, 2016)**,** have listed all the organizations and their standardization works related to IoT such as One M2M, Allseen Alliance, AIM (Association for Automatic Identification and Mobility), Eclipse Foundation, ETSI (European Telecommunications Standards Institute), and many more.

Universiti Malaya

2.3 IoT Home Application Development

Recently, IoT Home application has become a popular development application area among software developers who are interested in IoT software development. Home automation is known also as Smart Home. In the next few years, the demand of IoT Home applications and services will grow tremendously as the smart things and smart appliances increase in the market and are widely used domestically (Stojkoska & Trivodaliev, 2017). The term “Smart Home” has been a famous phrase searched at Google scholar with regards to IoT since 2004 (Stojkoska & Trivodaliev, 2017). Smart Things or smart objects in IoT home automation are described as controlling home appliances and features using information, communication, and sensing technology (Stojkoska & Trivodaliev, 2017).

Stojkoska and Trivodaliev (2017) show the Google search trend on IoT terms used in Figure 2.2.

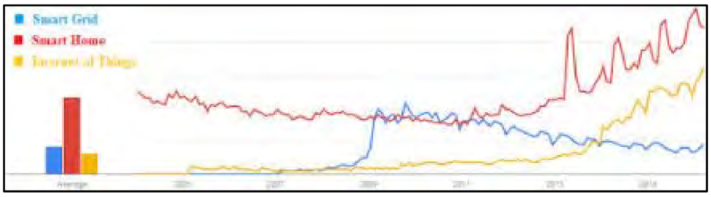


Figure 2.2 : Google search trend at Google scholar on IoT terms (Stojkoska & Trivodaliev, 2017)

Universiti Malaya

IoT Home application development has been developed using different sensing and control technologies such as RIFD, Bluetooth, Wi-Fi, and mobile networking (Piyare & Lee, 2013). (Piyare & Lee, 2013; Piyare, 2013) propose a system and architecture for smart home using Arduino microcontroller and Android mobile application for controlling and monitoring home appliances. Arduino Uno and Ethernet shield are used as the microcontroller and for internet connection. The connection used is Wi-Fi connection or mobile networks such as 3G or 4G. The proposed smart home in (Piyare & Lee, 2013; Piyare, 2013) is a flexible and low budget using the android smart phone to control smart devices at home via network communication. The study in (Kumar, 2014) presents a wireless smart home system using Android application, Arduino Mega 2560, Arduino Ethernet shield, and voice activation. The voice activation features added to their smart home app are especially for the use of the elderly and physically challenged users. The smart home app is to control and monitor devices via the internet and have a password change option.

Piyare and Lee (2013); Piyare, (2013) shows in Figure 2.3 the overview of the smart home architecture.

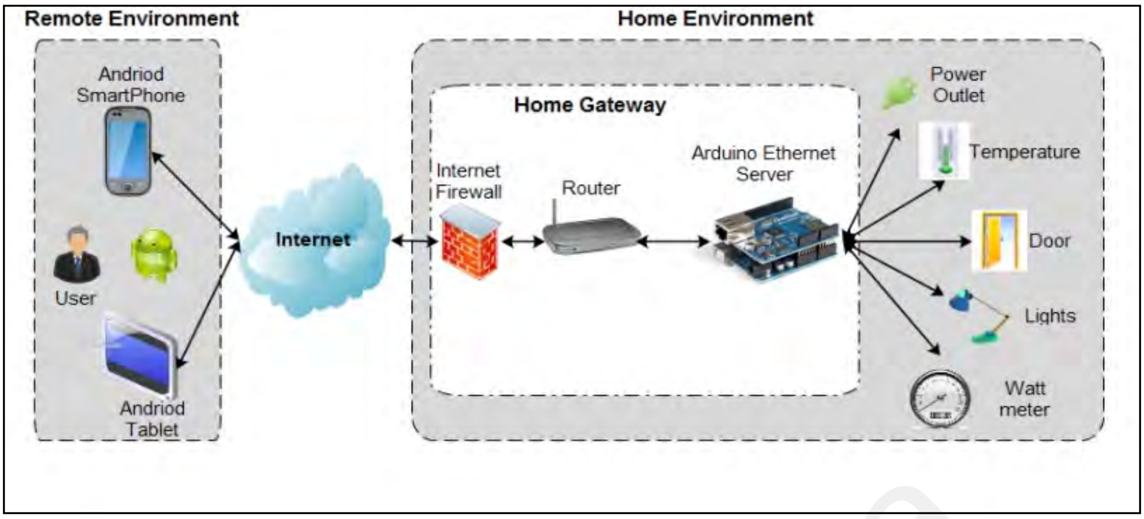


Figure 2.3 : Overview of the smart home architecture (Piyare & Lee, 2013; Piyare, 2013)

Furthermore, two models regarding home automation are introduced: 1) Home Automation via Bluetooth; and 2) Home Automation via the Internet connection (Abivandhana et al. 2017). The Home Automation via Bluetooth application involves Arduino Uno, Arduino Bluetooth, and Android mobile application to control home devices. Meanwhile, Home Automation via Internet Connection is designed using Arduino Uno, Arduino Ethernet shield, and Android mobile application and is based on details of IP address and Port number. In (Ghosh et al. 2015), a wireless home automation control system that connects through IP for communicating and controlling remotely house appliances and devices using Android mobile application and via Arduino Uno is presented. Apart from that, this home automation system has extra features such as voice control and reminders. Similar to the previously mentioned home automation system, the study in (Vinay & Kusuma, 2015) designs a home automation system with the Internet of Things except the data or information storage is with Cloud. IoT home automation via Arduino for IR devices is proposed by (Ismaeel & Kamal, 2017). This system is more to controlling devices with IR signals such as Television,

Universiti Malaya

DVD player and, Air conditioner. The proposed system by (Ismaeel & Kamal, 2017) is using Arduino Uno microcontroller, ESP8266 for the Wi-Fi connection, Cloud connection, the mobile application which includes iOS and Android as well as Ipad/Tablet.

A Home automation and security system was presented by (Wadhwani et al. 2018), utilizing mobile application, Arduino UNO and ESP8266 Wi-Fi Module as microcontrollers. The purpose for this home automation system is to sensing, control and monitor the opening and closing of door, and home appliances. Apart from this, the presented system able to alert on the door locks break and detecting fire. Besides, the authors in (Charumathi et al. 2018), have presented Android Home automation system which controls the on and off of the home appliances using Android mobile application and Arduino UNO as microcontroller through internet connection. Similarly, a smart home device employs Raspberry Pi microcontroller, web application and mobile application was proposed in (Reddy & Sankara, 2018). This work was designed to control light, air conditioner, fan and lights through Wi-Fi communication.

Table 2.2 shows a comparison study with regards to existing IoT Home application focused on development of IoT Home application.

Universiti Malaya

Table 2.2: Comparison study of existing IoT Home application focused on IoT home application development

Applications	Microcontroller	Communication Connection	Limitation	Mobile/Web-based
Smart Home-Control and Monitoring System	Arduino Uno and Ethernet shield	Wi-Fi connection, mobile networks (e.g. 3G, 4G)	Lack of guidelines on setting up hardware, Arduino board programming, and IoT mobile application. Focuses on Android mobile only for mobile application.	Android Mobile
Smart home system	Arduino Mega 2560, Arduino Ethernet shield	Internet connection	Lack of guidelines on setting up hardware, Arduino board programming, and IoT mobile application. Focuses on Android mobile only for mobile application.	Android Mobile
Home Automation using Bluetooth	Arduino Uno, Arduino Bluetooth	Bluetooth	Lack of guidelines on setting up hardware, Arduino board programming, and IoT mobile application. Focuses on Android mobile only the mobile application.	Android Mobile

Universiti Malaya

Table 2.2 continued

Home Automation using Internet connection	Arduino Uno, Arduino Ethernet shield	Internet connection	Lack of guidelines on setting up hardware, Arduino board programming, and IoT mobile application. Focuses on Android mobile only for mobile application.	Android Mobile
Wireless Home Automation	Arduino Uno	Internet connection	Lack of guidelines on setting up hardware, Arduino board programming, and IoT mobile application. Focuses on Android mobile only for mobile application.	Android Mobile
Home Automation System	Arduino Mega	Internet connection	Lack of guidelines on setting up hardware, Arduino board programming, and IoT mobile application. Focuses on Android mobile only for mobile application.	Android Mobile
IoT home automated system	ESP8266, Arduino Uno	Internet connection	Lack of guidelines on setting up hardware, Arduino board programming, and IoT mobile application.	Android, iOS Mobile, and Web-Based

Universiti Malaya

Table 2.2 continued

Home automation and security system	Arduino UNO, Wi-Fi Module, Accelerometer,	Internet connection	Lack of guidelines on setting up hardware, Arduino board programming, and IoT mobile application.	Mobile application
Android Home automation system	Arduino UNO	Internet connection	Lack of guidelines on setting up hardware, Arduino board programming, and IoT mobile application. Focuses on Android mobile only for mobile application.	Android Mobile
Smart home control device	Raspberry Pi	Internet connection	Lack of guidelines on setting up hardware, Arduino board programming, and IoT mobile application.	Mobile application & Web-based

Universiti Malaya

2.4 IoT software development approach

The term Software implies a rundown of machine directions where as the Engineering implies the utilization of disciplined approaches or methodologies and laws when building software systems (Ramachandran, 2012). Web of Things approaches are unquestionably encouraging and would probably represent foundation technologies in the future of IoT (Zambonelli, 2016). The author in (Zambonelli, 2016) elaborated that along Web of Things there are, various diverse methodologies such as programming approaches and approaches supporting middleware are being proposed to help the designing of IoT applications and systems. However, Zambonelli (2016) further explained that there is still a lack of a common unifying approach to design and development based on a common set of abstractions, models and methodologies and this undermines the likelihood of advancing an efficient and disciplined methodology towards the development of complex IoT systems, and therefore restricts unfolding the true abilities of the IoT vision. Developers integrate and compose intelligent devices and software applications when developing smart home systems. Due to their diversity and heterogeneity, developers usually face many difficulties (Wu et al. 2017). The authors in (Wu et al. 2017) presented a model-based approach to the development of smart home systems as a solution to the problem faced by the developers and stated that the model-driven approach will facilitate the development of smart home systems for developers.

Many IoT research efforts have been devoted to device, networking and application service perspectives to date, software engineering approaches for the development of IoT systems are still in their early stages (Fortino et al. 2015). Fortino et al. (2015) presents a new approach to software engineering to support the systematic development of Smart Objects based systems and the approach proposed is based on metamodels defined at different levels of abstraction to support the analysis, design and implementation phases.

The development of applications in the IoT is challenging because it addresses a wide range of related issues, such as the lack of separation of concerns and the lack of high abstractions to address both large-scale and heterogeneous issues (Patel & Cassou, 2015). In addition, stakeholders involved in the development of applications must address issues that can be attributed to different life cycles in the development of applications (Patel & Cassou, 2015). The authors in (Patel & Cassou, 2015) stated

Universiti Malaya

several approaches in the closely related fields of wireless sensor networks, ubiquitous and extensive computing and software engineering in general have been proposed to address the above challenges. However, existing approaches cover only limited subsets of the challenges mentioned above when applied to the IoT system. As such, the authors (Patel & Cassou, 2015) presented a methodology for the development of IoT applications based on techniques in the fields of sensor network, macro programming and model-driven development which divides the development of IoT applications into different issues and integrates a set of high-level languages.

A disciplined and rigorous approach to software analysis, design and development is required to ensure that IoT services and applications are designed and developed in a reliable manner (Zambonelli, 2016). In a different article, Zambonelli (2016) stated that in spite of the fast progresses in IoT study, a common principled software engineering approach for the efficient development of IoT systems and applications is still missing. Ramachandran (2012) described that guidelines help to determine an appropriate design criterion when it comes to design decisions. Software guidelines therefore summarize expert knowledge as a collection of judgments, rationales and principles of design. The author further stated that guidelines provide the foundation for a solution that has worked well in previous applications and the environment. Engineers can use guidelines to learn new principles with examples and experts alike (Ramachandran, 2012).

Universiti Malaya

2.5 Guideline-based development approach for IoT development process

Software development process and methodologies are combinations of methods, procedures, development kits, activities, and principles that guide the developers in developing a software system. Software development approaches or methodologies are guidelines that direct developers on what to be done, how and when it ought to be done throughout the software development process (Klimes & Prochazka, 2006). The development approach also helps us not to overlook anything, picture us with what the outcome should be, and furthermore, what not to be developed or be done in a specific stage of the development (Klimes & Prochazka, 2006). Guidelines give an exact set of steps in light of fundamental software design principles which offer assistance for us to take any course of an efficient set of activities (Ramachandran, 2012).

The word guideline is defined as follows in (Ramachandran, 2012):

- A suggested approach, parameter, etc. for carrying out an activity or assignment, using a product, etc.; and
- An assertion of required, good or best practice.

Software guidelines outline professional knowledge as a set of methods of reasoning, design judgments, and standards. Many guidelines for IoT Home application are accessible in different sources, for example, IoT based websites, journal articles, and books. Most of the sources do not cover or provide guidelines throughout the development process but only for one or few phases of the development. The guidelines that are presented at the resources come in different categories such as: best practices.

- Best Practices

A best practice is characterized as a process or practice that has been for the most part acknowledged as better than any options since it produces outcomes that are better than those accomplished by different means or in light of the fact that it has turned into a standard method for getting things done (Rouse, 2007).The term best practice has maintained the information and knowledge that become known since a long time ago of fruitful use over a numerous system, software projects, products, and software development applications (Ramachandran, 2012). For the most part, guidelines are

Universiti Malaya

given (generally in textual outline) for software engineers to accentuate on a few practices that reinforce the nature of software systems, and caution them from a few consequence that debilitates the results of the software systems. Best practices give a well-ordered instruction or solution to computer program issue over the life cycle and depend on the effective utilization in the true world (Ramachandran, 2012).

An example of good practices that supports global software development, which was recognized in the course of recent years (Ebert & De Neve, 2001) includes:

- The project aim or outcome such as objective, quality, resource allocation, is discussed and agreed at the earlier stage of the project.
- Elect one team project manager who is completely accountable for accomplishing the project aim.
- An interactive development model to be presented in light of acknowledged good practices that permit tailoring processes for the particular requirements of a team or project.
- Come up with the project page that bridges project content, the project duration, information of the planning and design of the project.

Universiti Malaya

A list of best practices is proposed by (Gyrard et al. 2015) to additionally improve the publication and finding of ontology in the IoT area. The list of the best practices proposed are divided into five classes: 1) ontology & dataset publication (formats, serialization) - four best practices, 2) metadata for vocabulary - two best practices, 3) ontology quality - three best practices, 4) ontology & dataset reuse - four best practices, and 5) namespace management - three best practices (Gyrard et al. 2015). An example of the proposed best practices is: “Best Practice 1. The format used to represent IoT data should be unified and we should provide wrappers/translators to deal with heterogeneous formats (e.g., SenML/XML, SenML/JSON, CSV, Excel)” (Gyrard et al.

2015, p. 3), and “Best Practice 2. The format used to represent semantic IoT data (e.g., RDF/XML, JSON-LD, Turtle, Ntriples) should be unified to ease the interlinking of datasets.” (Gyrard et al. 2015, p. 3).

IBM Corporation has come out with an article written by Fisher (2015) on the best practices for IoT development. The mobile application best practices are proposed and presented to be applied as best practices for IoT development (Fisher, 2015). The best practices proposed for IoT Development are: “Assume that the network can disappear at any moment - even (and especially) midway through a transfer” (Fisher, 2015, p.4), “Use smaller messages, more frequently, rather than single, large requests and responses” (Fisher, 2015, p.4), and “Ensure that the application assesses connectivity, and don't expect the user to be aware of it” (Fisher, 2015, p.4).

Universiti Malaya

2.6 Proposed Guideline-based Development Approach for IoT Home Applications

As discussed in the previous sections, guidelines are a required aspect of the IoT Home application software development process for software developers.

Within this research, a new guideline-based development approach is proposed to guide software developers throughout development stages while developing software systems. The new guideline-based development approach is to be applied from the early stage of the development till the end stage of the development process. The guideline-based development approach will be presented in subsequent chapters in form of steps by step approach, diagrams, sample coding, patterns, checklists and best practices throughout the software development phases for the IoT home application.

The proposed guideline-based development approach provides software developers with an appropriate consistent set of guidelines at each development phase that are to be applied while developing an IoT home application, to ultimately guide the developers in producing a fully completed system within the stipulated time.

2.7 Chapter Summary

Guidelines contribute to the development process of IoT Home application system, in hardware and software development stages. The guidelines are not delimited to one software development stage but should be considered in all the stages of the software development process.

To develop IoT Home application systems, there are several guidelines accessible in different sources: websites, journal articles, books, etc with a variety of forms, including in sequence order steps, UML diagram, design patterns and best practices. However, the guidelines are incomplete and no one place that includes all together the required guidelines so as to be alluded to by software developers. Therefore, software developers might not be able to discover appropriate guidelines to refer while developing an IoT Home application system because of stipulated time, doesn't have access to the required reliable source and less of expertise on searching for the related guidelines. Due to this, the development process of IoT Home application system will be affected.

Apart from this, hardware development stage is a challenge for software developers as they may not have minimal experience in setting up the hardware or smart devices and the programming part for the hardware. This research intends to guide the software developers so that they do not need to depend on others such as the electrical and electronic engineers. Furthermore, choosing a programming language to use for IoT

Universiti Malaya

Home application development can be as big a decision as choosing a hardware platform. Some might be effective for large-scale systems, while some might be for small-scale systems. The programming language can be chosen based on certain criteria such as the operating system, software development tools used in the development, hardware devices, and chosen open source.

Having said that, the present state of the art does not present guidelines to help and direct software developers on developing IoT Home application system from the early stage till the last stage of the development process. As such, this research intends to propose guideline-based development approach for IoT home application.

Universiti Malaya

CHAPTER 3: RESEARCH METHODOLOGY

3.1 Introduction

The goal of this research is to propose a guideline-based development approach for IoT Home application. The proposed approach is referred as guidance and applied in the process of IoT Home application development from the early stage until the end of the development stage.

The approach for this research has been done in a set of steps throughout five stages, as depicted in Figure 3.1: 1) Identification of the problem stage; 2) Analysis for solution stage; 3) Proposing guideline-based development approach; 4) System design, and development stage; and 5) Evaluation stage. The identification of the problem stage starts with a background study on the literature of IoT, IoT related to Software engineering and software development domain. The literature study is continued by going through the previous and related work on IoT, development related to the IoT as well as software engineering. During the study, the initial research objective and questions were identified. After that, the second stage continued with the analysis of a solution. At this stage, the literature study is continued on the software development related to IoT and further narrowed down to IoT Home application development approach. An interview session was also conducted with IoT developer and expert on the IoT Home application development process. Moreover, the solution for the identified problem of IoT Home application development approach is defined. The initial objectives and questions stated in stage 1 were reanalyzed.

At stage 3 of this research, a guideline-based development approach is proposed for IoT Home application development. The guideline-based development approach is involves different categories such as patterns, UML diagrams, sequence of steps, checklist, and best practices. The fourth stage of this research is on system design and

Universiti Malaya

development. IoT Home application system is developed by referring and applying the guideline-based development approach defined at the third stage of this research.

The final stage of this research is on evaluation. The guideline-based development approach for IoT Home application and the developed system are evaluated by experts and users. The evaluation is conducted based on usability. The details and result of the evaluation are analyzed and documented.

The five stages of research methodology are discussed in more detail in the following sections.

Universiti Malaya

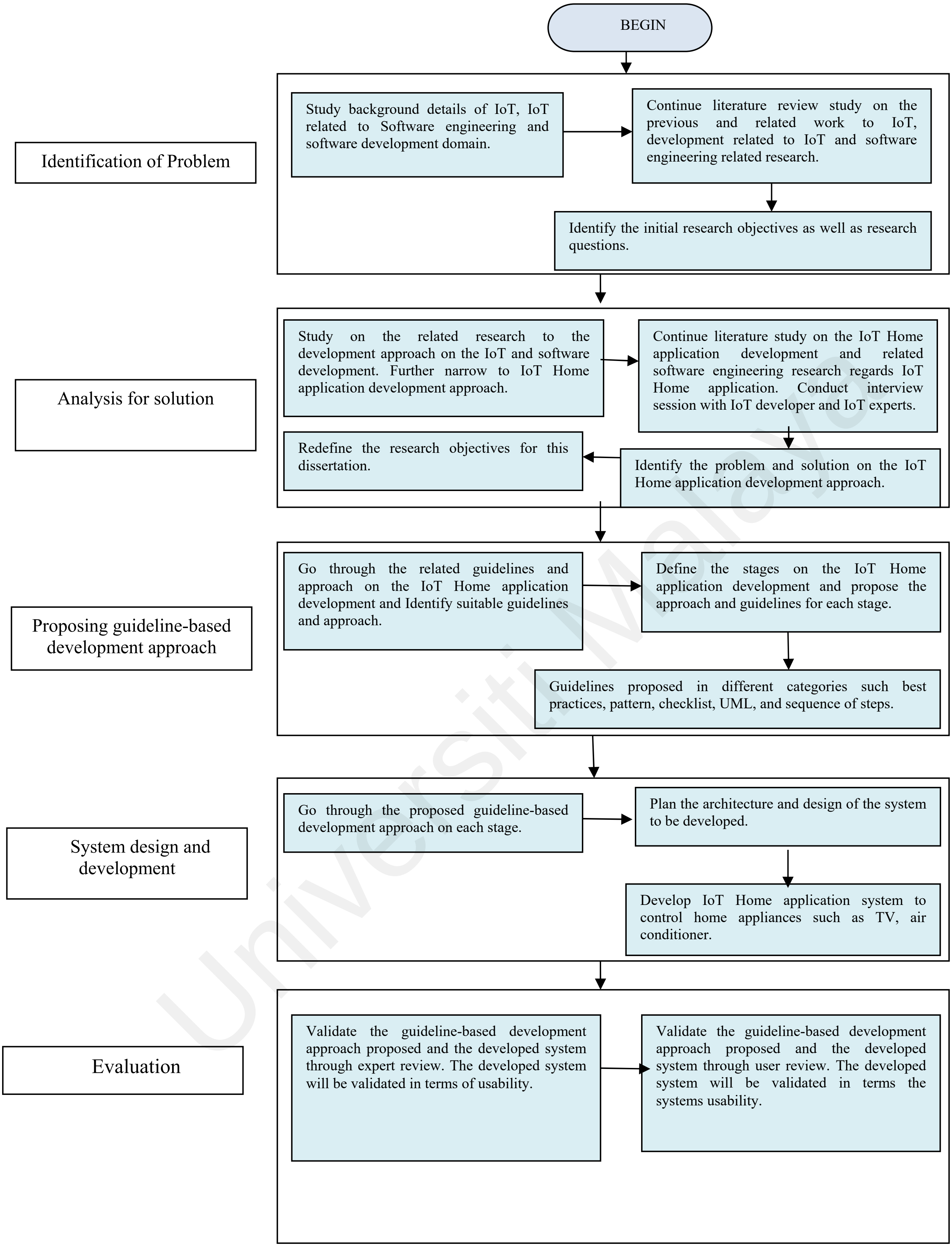

BEGIN
Identification of Problem
Study background details of IoT, IoT related to Software engineering and software development domain.
Continue literature review study on the previous and related work to IoT, development related to IoT and software engineering related research.
Identify the initial research objectives as well as research questions.
Analysis for solution
Study on the related research to the development approach on the IoT and software development. Further narrow to IoT Home application development approach.
Continue literature study on the IoT Home application development and related software engineering research regards IoT Home application. Conduct interview session with IoT developer and IoT experts.
Redefine the research objectives for this dissertation.
Identify the problem and solution on the IoT Home application development approach.
Proposing guideline-based development approach
Go through the related guidelines and approach on the IoT Home application development and Identify suitable guidelines and approach.
Define the stages on the IoT Home application development and propose the approach and guidelines for each stage.
Guidelines proposed in different categories such best practices, pattern, checklist, UML, and sequence of steps.
System design and development
Go through the proposed guideline-based development approach on each stage.
Plan the architecture and design of the system to be developed.
Develop IoT Home application system to control home appliances such as TV, air conditioner.
Evaluation
Validate the guideline-based development approach proposed and the developed system through expert review. The developed system will be validated in terms of usability.
Validate the guideline-based development approach proposed and the developed system through user review. The developed system will be validated in terms the systems usability.

Figure 3.1 : Research Methodology

Universiti Malaya

3.2 Identification of problem stage

At this stage, a background study is conducted on IoT, Software Engineering, and software development domain. Early research objectives and questions were recognized. This stage is further elaborated in the following manner:

3.2.1 Background study on IoT, Software Engineering and Software Development

First of all, a background study was conducted on the following domains:

1. IoT;
2. IoT related to Software engineering; and
3. IoT related to Software development.

The literature study was conducted by searching academic databases for the above-mentioned terms. The results from the search were downloaded and read through. The academic databases searched involved: Google Scholar, IEEE Xplore, ScienceDirect, ACM Digital Library, ResearchGate, and Elsevier.

IoT is growing rapidly and a hot topic for researchers all over the world. There are a lot of research that are completed and still in progress on IoT related to software development and software engineering. At this stage, IoT background was studied from reliable conference papers (e.g. Khan et al. 2012), journal articles (e.g. Gubbi et al. 2013), and several reliable websites (e.g. Eclipse IoT Website, 2015).

The study on the IoT includes the definition, characteristics of IoT, and IoT features. Furthermore, IoT application areas such as the industrial area, health, home automation and etc were focused while studying the IoT background. Apart from that, the

Universiti Malaya

background study continued on the IoT challenges such as security wise, IoT architecture, the research challenges in IoT and the future direction of IoT.

As for the study on the IoT related software engineering and development, the following topics were focused: -

- current IoT tools available for IoT development
- IoT open source and standards
- IoT technologies
- IoT development application area
- IoT methodology, and
- Characteristics of IoT software projects.

3.2.2 Study on previous and related research on IoT Software engineering and Development associated with IoT

There are a lot of previous and current research articles related to IoT Software Engineering and development available at journal websites. These articles are further studied in order to identify the problem for this research. The articles provide solutions with regards to the software engineering and development area. There are different kinds of solution presented, such as frameworks to help developers to develop the IoT system and to solve the security-related problem. Other than that, the researchers also come out with new IoT tools for the software development process, IoT programming models, IoT programming approach, IoT architecture, and new IoT technologies for the IoT software development as a solution for their identified problems.

Universiti Malaya

Apart from articles, there are also popular websites such as Eclipse IoT Website involved in discussing and introducing new technologies and tools as well as an open source for IoT development.

3.2.3 Identify the initial research objectives and questions

At this step of this stage, the initial research aim or objective, as well as research questions were identified. Following are the research objectives and questions:

Universiti Malaya

Research objectives

1. To identify the problems and challenges faced by the software developers on the IoT application development process;
2. To propose a guideline-based development approach for software developers in order to ease the IoT development process;
3. To design a prototype for the IoT application based on the proposed guideline-based development approach; and
4. To evaluate the prototype of the IoT application based on the users and experts evaluation in terms of performance and usability as well as the proposed guideline-based development approach.

Research questions

1. What are the problems and challenges faced by the software developers on the IoT application development process?
2. What is the guideline-based development approach for software developers that could help to ease the IoT development process?

3. In what forms or categories the guidelines will be proposed for IoT application development approach?
4. How to evaluate the proposed guideline-based development approach and the developed prototype?

3.3 Analysis for solution stage

In the second stage, identification of the research problem and solution for the identified problem is continued by examining the literature and conducting interviews. Literature study proceeded on the IoT software development and further limited to IoT Home application development. An interview was conducted with interviewing IoT developer and IoT expert's regards to IoT Home application development. Furthermore, the solution is recognized for the problem that was identified with regards to IoT Home application development. The initial objectives and questions were redefined and a set of research objectives and questions were formulated.

3.3.1 Literature study on IoT software development and IoT Home application development

A comprehensive review of the literature was conducted based on various articles of IoT software development (e.g. Gubbi et al. 2013, Hiesgen, 2015, Patel et al. 2016). Moreover, the study further narrowed to IoT Home application development by studying the related articles (e.g. Stojkoska & Trivodaliev, 2017, Piyare & Lee, 2013, Kumar, 2014). This study's research problem was addressed after performing a careful reviewing and analyzing the relevant literature.

Universiti Malaya

3.3.2 Interview approach on IoT Home application development

An interview approach is used to gather information to identify the problem and feedback on the recommended solution related to IoT Home application development. The data acquired from the interviews with IoT developer and IoT experts derived to identification of research problem in the development process and suitable solution which is guideline-based development approach. The interview questions were focused on the problems faced by the new developers and IoT developers throughout the development process, the details on the IoT Home application development, development stages or phases in IoT Home application and what are recommended solution by the interviewee. The interview question is included in Appendix B.1: Expert interview questions and the respondent's answer included in Appendix B.2: Expert's Interview Answers.

We have interviewed three experts who are:

1. One IoT developer who are also a Phd student in University Malaya and have involved with multiple IoT projects in University Malaya; and
2. Two IoT experts who are involves in smart city project in Malaysia.

Universiti Malaya

3.3.3 Solution identification for the research problem defined

Given that the research problem identified, the literature study continued on searching for the suitable solution**.** A guideline-based development approach was proposed as a suitable solution for the IoT Home application development problem.

3.3.4 Redefined research objectives

In this second stage, the initial research objective and research questions were reanalyzed. Following that, a set of research objectives and questions that were formulated:

- Research objective 1: To identify the problem and challenges faced by the software developers on the IoT Home application development process.
- Research objective 2: To identify the available guidelines for the IoT home application development.
- Research objective 3: To propose a guideline-based development approach for developers includes the requirement review stage till the mobile application stage in order to improve the IoT Home application software development process.
- Research objective 4: To develop a prototype for the IoT Home application for controlling the Home appliances by applying the proposed guideline-based development approach.
- Research objective 5: To validate the proposed guideline-based development approach and the prototype based on the user evaluation and expert review.

Universiti Malaya

3.4 Proposing guideline-based development approach

The overall development stages of the IoT Home application were identified. At each development stage of the IoT Home application, different tasks are performed. Therefore at each development stage, guidelines required to be applied to the needed tasks are different, in order to develop a quality IoT Home application system. Moreover, guidelines are available at all kinds of sources but not all can be a reliable source and able to access them. Thus, the required guidelines were gathered and composed from reliable sources such as articles and conference papers and reliable journals, guideline brochures from well-known companies, and reliable websites such as IBM, Microsoft.

As a large number of guidelines are available, the suitable guidelines for each of the development stages were identified.

The identified guidelines were divided into the respective IoT Home application development stages. The guidelines come in the form of best practices, pattern, diagrams, checklist, and sequence order steps. The guidelines were also proposed for the requirement review process, development environment preparation process, hardware setting up process, choosing suitable programming language process, Arduino board coding process and mobile application development process. The best practices were proposed for all the stages except for hardware setting up stage. Meanwhile, sequence order steps were also proposed for all development processes except choosing programming language process. A guideline-based development approach application was developed to assist IoT developers in developing the IoT Home application.

Universiti Malaya

3.5 System design and development stage

Given that the required approach have been proposed for the development of IoT Home application in the earlier stage, this research continued to the system design and development stage. At this stage, the IoT Home application system was developed by applying the proposed guideline-based development approach. The development of the IoT Home application has started by planning, analyzing and designing the IoT Home application system. The guidelines proposed for the requirement review process is referred and applied. After that, the development environment was prepared and continued with the hardware set up by applying the guideline proposed in the development environment preparation process and hardware set up process. Next, the coding for the Arduino board and the mobile application performed. The development for Arduino and mobile application is conducted by referring to the guideline proposed in Arduino board programming process, choosing a suitable programming language process and mobile application development process. Finally, the testing and the deployment stage were carried out.

Universiti Malaya

3.6 Evaluation stage

This research was evaluated on different levels. The proposed guideline-based development approach was evaluated using expert review whereby a group of IoT and software development experts were asked to provide their opinions and reviews on the proposed guidelines of this research. The expert review also is done on the developed IoT Home application system which was developed by referring to the proposed approach. The guideline-based development approach was also validated by a group of users who are software developers through user evaluation. The user evaluation was conducted on the developed IoT Home application system in terms of its usability. The expert and user evaluation is carried out using questionnaire approach. The experts and users who evaluate are required to answer either ‘yes’, ‘no’ or ‘not sure’. The experts and users are welcomed to comment on each question answered. Five experts and twenty users have taken part in this evaluation process. The proposed guideline-based development approach’s effectiveness and convenience and the usability of the developed IoT Home application was evaluated. The guideline-based development approach was explained and presented before the evaluation was done. The IoT Home application was also demonstrated and presented as well before the user evaluation process was carried out. The evaluation stage is presented in detail in Chapter 6.

3.7 Chapter Summary

This study proposes a guideline-based development approach to be applied while developing an IoT Home application system, for the reason of helping software developers to produce a quality system within a required time. This research was carried out in five main stages: 1) Identification of the problem stage, where the literature study on the IoT and software engineering and development background was performed; 2) analysis stage for solution, continued with comprehensive literature study on the IoT software development and further limited to IoT Home application development to identify the solution for the problem; 3) proposing approach, propose a guideline-based development approach for the IoT Home application; 4) system design and development stage, the proposed guidelines were applied in developing the IoT Home application system; and 5) evaluation stage, where the proposed approach and the developed system were evaluated.

The next chapter will discuss further on the proposed guideline-based development approach for IoT Home application.

Universiti Malaya

CHAPTER 4: GUIDELINE-BASED DEVELOPMENT APPROACH FOR IOT HOME APPLICATIONS

4.1 Introduction

The proposed guideline-based development approach aims to assist software engineers throughout the IoT Home application development process. As mentioned in the previous chapter, the guidelines for the IoT Home application development is available in various sources but there is no one source that provides the development guidelines from the early stage: requirement review until the last stage: mobile application development for IoT Home application development process. Software developers might have a tough time on finding suitable guidelines to allude while developing as it is scattered over different sources, fixed time frame constraint, limited access to necessary guideline sources, and less proficiency in searching for related guidelines.

Besides, software developers might not have the required experience and knowledge on setting up the hardware part for the IoT Home application and this would lead software developer to depend on others who have the knowledge and experience such as electrical and electronic engineers. Furthermore, choosing the right programming language can be an as enormous choice as choosing hardware platform. Providing the guidelines to choose based on certain criteria rather than choosing based on knowing the programming language is important. Consequently, a guideline-based development approach is needed for IoT Home application development process.

This chapter presents the details of the proposed guideline-based development approach for developing IoT Home applications. IoT Home application development process is divided into six stages, which are requirements review process, development environment preparation process, hardware setup process, Arduino board programming

Universiti Malaya

process, mobile application development process, and testing process. The six stages are defined as illustrated in Figure 4.1. At each stage except for the testing stage, guidelines are proposed and presented in detail in suitable and task-supported forms:1) best practices, 2) patterns, 3) sequence of steps, and 4) checklist. The details of the guideline-based development approach are discussed in the following sections. Figure 4.2 shows the guideline-based development approach main page.

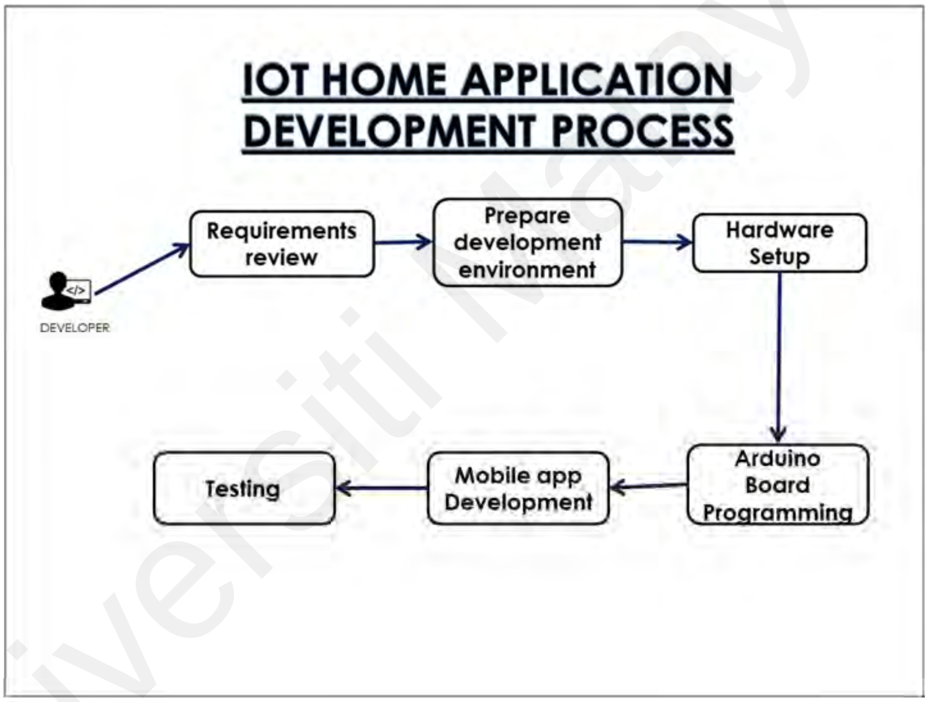
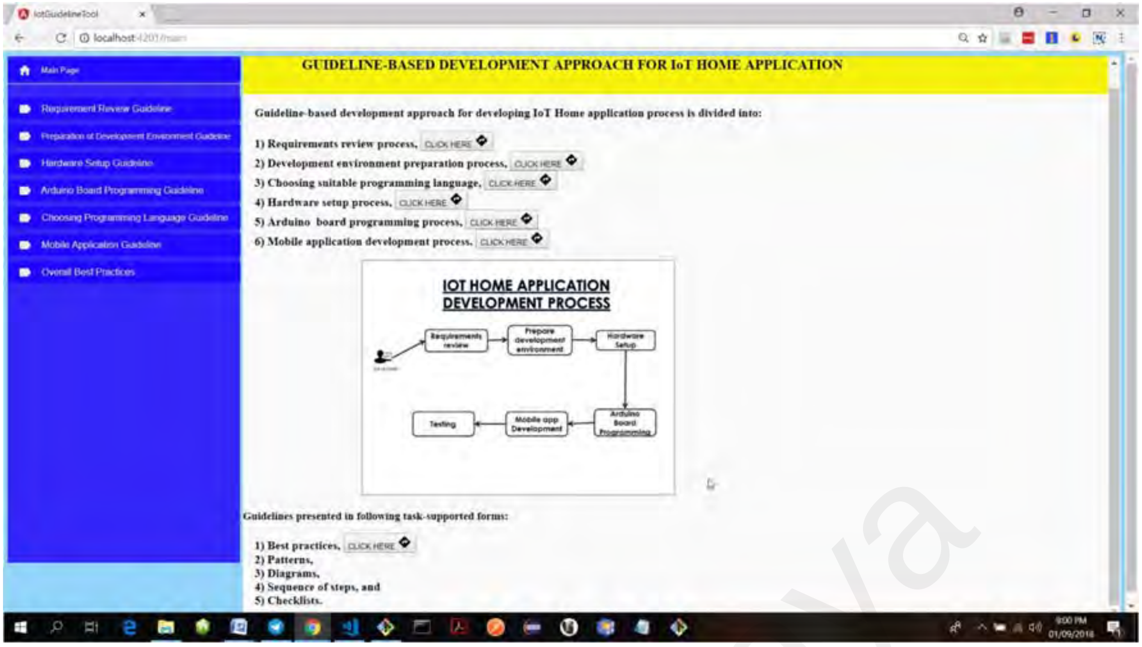

IOT HOME APPLICATION DEVELOPMENT PROCESS
DEVELOPER
Requirements review
Prepare development environment
Hardware Setup
Arduino Board Programming
Mobile app Development
Testing

Figure 4.1: The IoT Home application development process

Universiti Malaya


IotGuidelineTool
localhost:4200/main
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices
IOT HOME APPLICATION DEVELOPMENT PROCESS
Requirements review
Prepare development environment
Hardware Setup
Arduino Board Programming
Mobile app Development
Testing

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

Guideline-based development approach for developing IoT Home application process is divided into:

1) Requirements review process, CLICK HERE
2) Development environment preparation process, CLICK HERE
3) Choosing suitable programming language, CLICK HERE
4) Hardware setup process, CLICK HERE
5) Arduino board programming process, CLICK HERE
6) Mobile application development process. CLICK HERE

Guidelines presented in following task-supported forms:

1) Best practices, CLICK HERE
2) Patterns,
3) Diagrams,
4) Sequence of steps, and
5) Checklists.

Figure 4.2 : The guideline-based development approach main page

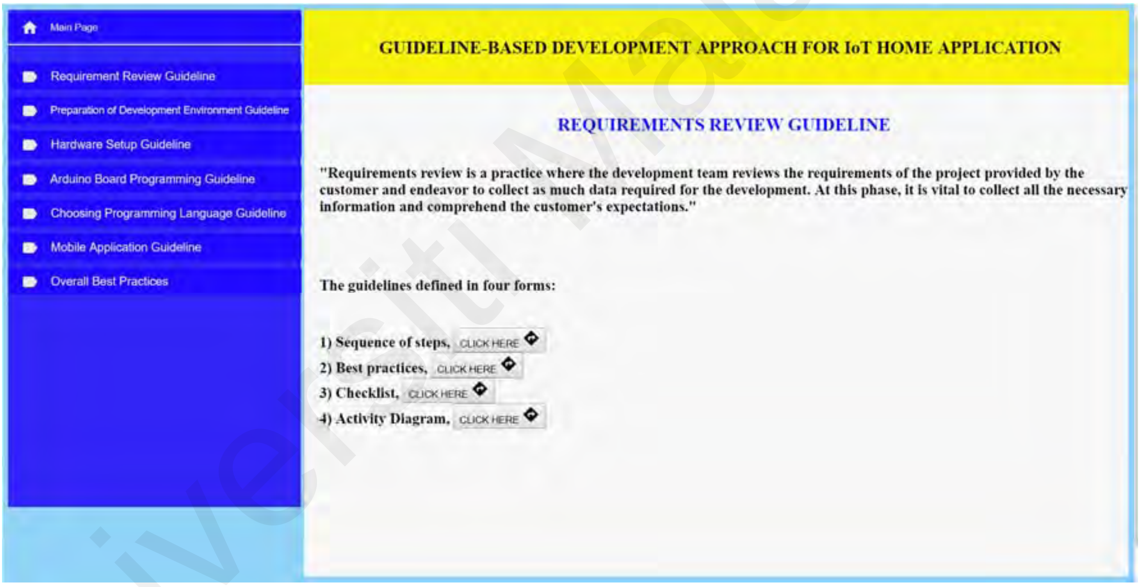
Universiti Malaya

4.2 Requirements review process

Requirements gathering and review process in the IoT Home application development where the development team looks upon the requirements of the client and attempt to accumulate as much information on the requirements required by the client. At this stage, it is important to gather all the required information, and understand the client's or stakeholders expectations. Apart from that, necessity to ensure that there is nothing important information required for the IoT Home application development is left out. Requirements review is carried out to avoid problems such as unclear stated requirement details at the later development stage and requirement conflict. Requirement review process is an early stage of validation for the new system to be developed and is considered as an information collection and quality assurance process. As such, guidelines at this stage are required to assist software developers in making certain that requirements are complete and the necessary information is available.

- **Guidelines for requirements review process**

The guidelines that are proposed for this process is defined in three forms: (i) sequence of steps, (ii) best practices, (iii) checklist, and (iv) activity diagram. Figure 4.3 shows the approach screen on requirements review main page. The sequence of steps details the steps needed to be attended in requirement review stage. The best practices and checklist are adapted while following the steps listed in the sequence of steps. The sequence of steps is represented in the activity diagram.


Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

REQUIREMENTS REVIEW GUIDELINE

"Requirements review is a practice where the development team reviews the requirements of the project provided by the customer and endeavor to collect as much data required for the development. At this phase, it is vital to collect all the necessary information and comprehend the customer's expectations."

The guidelines defined in four forms:

1) Sequence of steps, CLICK HERE
2) Best practices, CLICK HERE
3) Checklist, CLICK HERE
4) Activity Diagram, CLICK HERE

Figure 4.3 : The approach screen on requirements review guideline main page

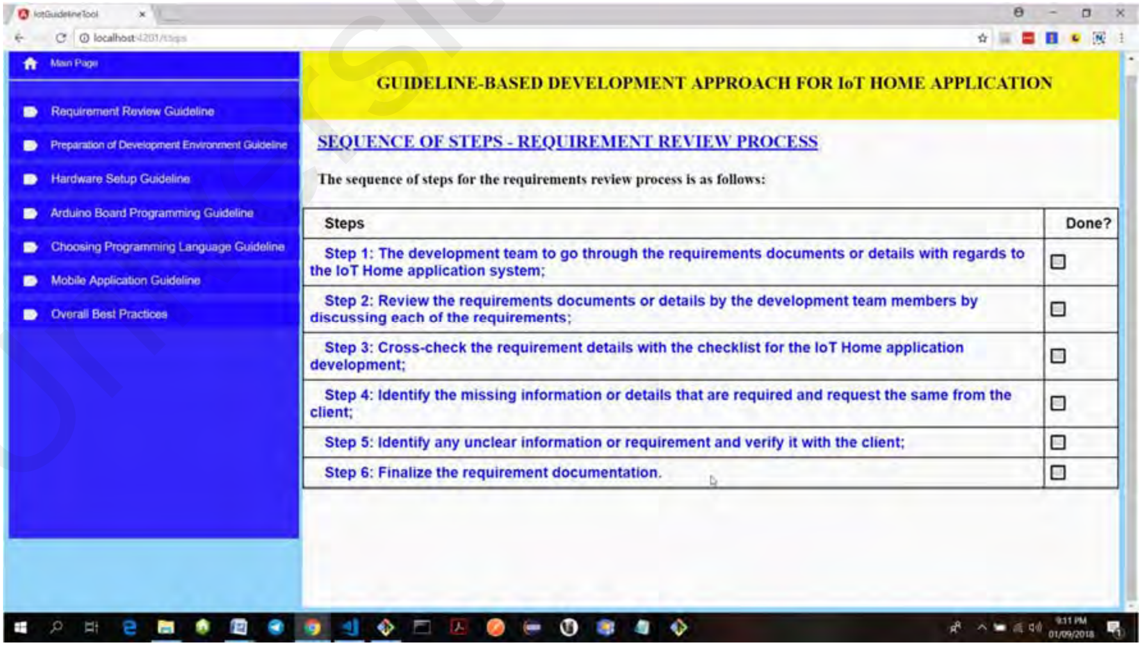
Universiti Malaya

The sequence of steps for the requirements review process is as follows:

- Step 1: The development team to go through the requirements documents or details with regards to the IoT Home application system;
- Step 2: Review the requirements documents or details by the development team members by discussing each of the requirements;

- Step 3: Cross-check the requirement details with the checklist for the IoT Home application development. Figure 4.5 and Figure 4.6 show the approach screen on requirements review checklist page. Figure 4.7 illustrates the requirements review checklist for the IoT Home application.
- Step 4: Identify the missing information or details that are required and request the same from the client;
- Step 5: Identify any unclear information or requirement and verify it with the client; and
- Step 6: Finalize the requirement documentation.

Figure 4.4 illustrate the approach screen related to requirements review process of sequence of steps.


IotGuidelineTool
localhost:4201/steps
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

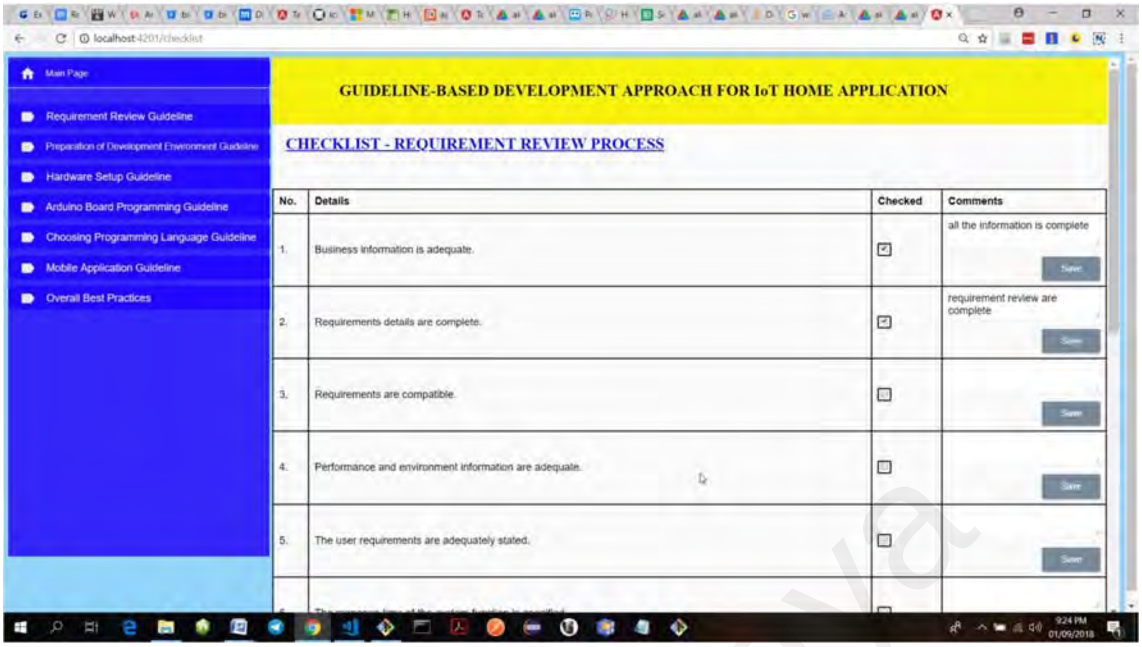
SEQUENCE OF STEPS - REQUIREMENT REVIEW PROCESS

The sequence of steps for the requirements review process is as follows:

Steps	Done?
Step 1: The development team to go through the requirements documents or details with regards to the IoT Home application system;	□
Step 2: Review the requirements documents or details by the development team members by discussing each of the requirements;	□
Step 3: Cross-check the requirement details with the checklist for the IoT Home application development;	□
Step 4: Identify the missing information or details that are required and request the same from the client;	□
Step 5: Identify any unclear information or requirement and verify it with the client;	□
Step 6: Finalize the requirement documentation.	□

Figure 4.4 : The approach screen on sequence of steps of requirements review process

Universiti Malaya


localhost:4201/checklist
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices
Save
9:24 PM
01/09/2018

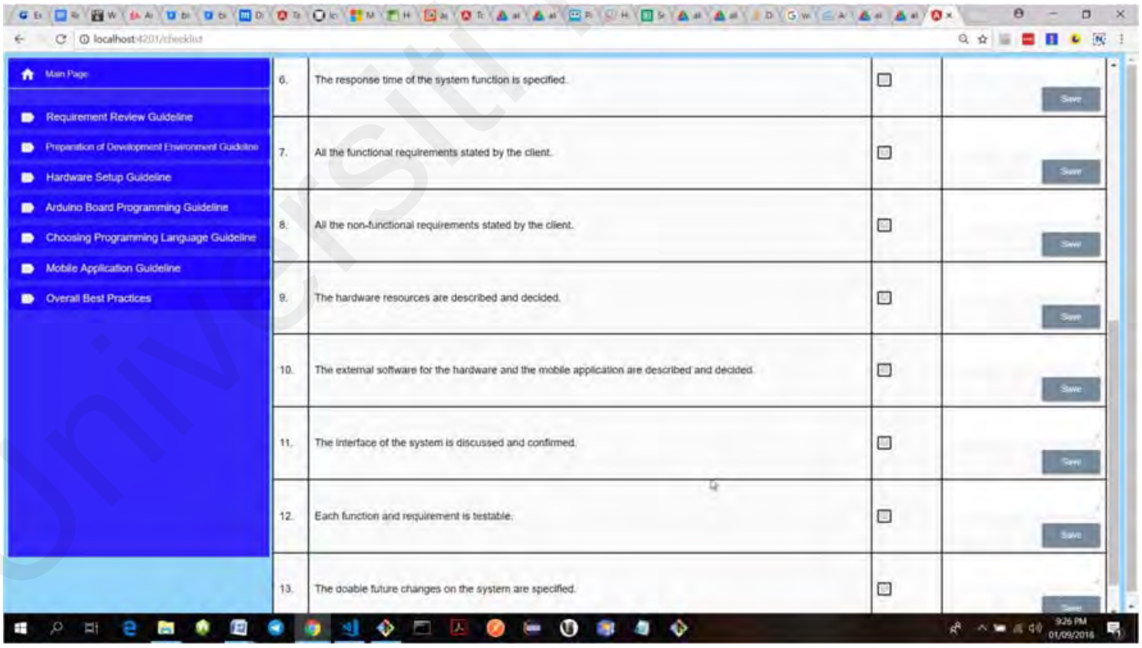
GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

CHECKLIST - REQUIREMENT REVIEW PROCESS

No.	Details	Checked	Comments
1.	Business information is adequate.	☑	all the information is complete
2.	Requirements details are complete.	☑	requirement review are complete
3.	Requirements are compatible.	□	
4.	Performance and environment information are adequate.	□	
5.	The user requirements are adequately stated.	□	

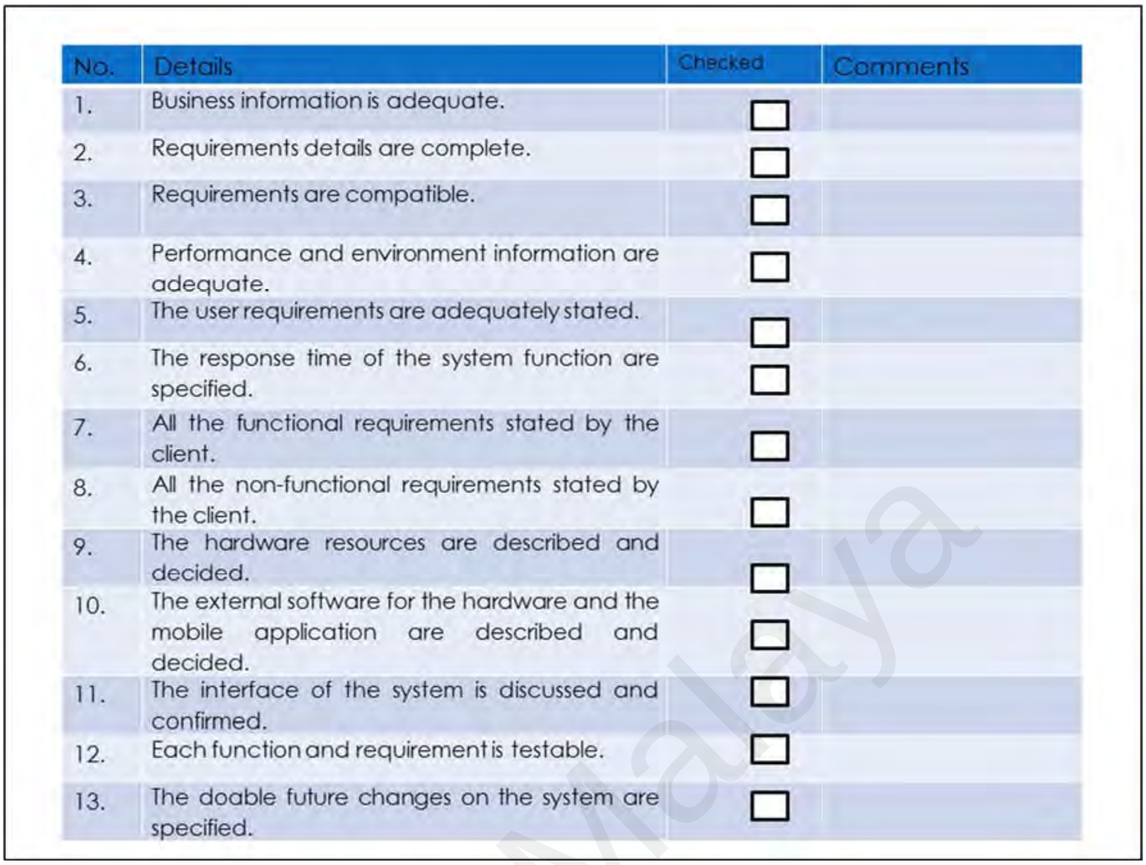
Figure 4.5 : The approch screen on checklist of requirements review process

Universiti Malaya


localhost:4201/checklist
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices
Save
9:26 PM
01/09/2018

6.	The response time of the system function is specified.	□	
7.	All the functional requirements stated by the client.	□	
8.	All the non-functional requirements stated by the client.	□	
9.	The hardware resources are described and decided.	□	
10.	The external software for the hardware and the mobile application are described and decided.	□	
11.	The interface of the system is discussed and confirmed.	□	
12.	Each function and requirement is testable.	□	
13.	The doable future changes on the system are specified.	□	

Figure 4.6 : The approch scroll screen on checklist of requirements review process



No.	Details	Checked	Comments
1.	Business information is adequate.	□	
2.	Requirements details are complete.	□	
3.	Requirements are compatible.	□	
4.	Performance and environment information are adequate.	□	
5.	The user requirements are adequately stated.	□	
6.	The response time of the system function are specified.	□	
7.	All the functional requirements stated by the client.	□	
8.	All the non-functional requirements stated by the client.	□	
9.	The hardware resources are described and decided.	□	
10.	The external software for the hardware and the mobile application are described and decided.	□	
11.	The interface of the system is discussed and confirmed.	□	
12.	Each function and requirement is testable.	□	
13.	The doable future changes on the system are specified.	□	

Figure 4.7 : The requirements review checklist

Universiti Malaya

Figure 4.8 shows an activity diagram for requirements review process involved in IoT Home application development and figure 4.9 shows the approach screen on activity diagram of requirements review process.

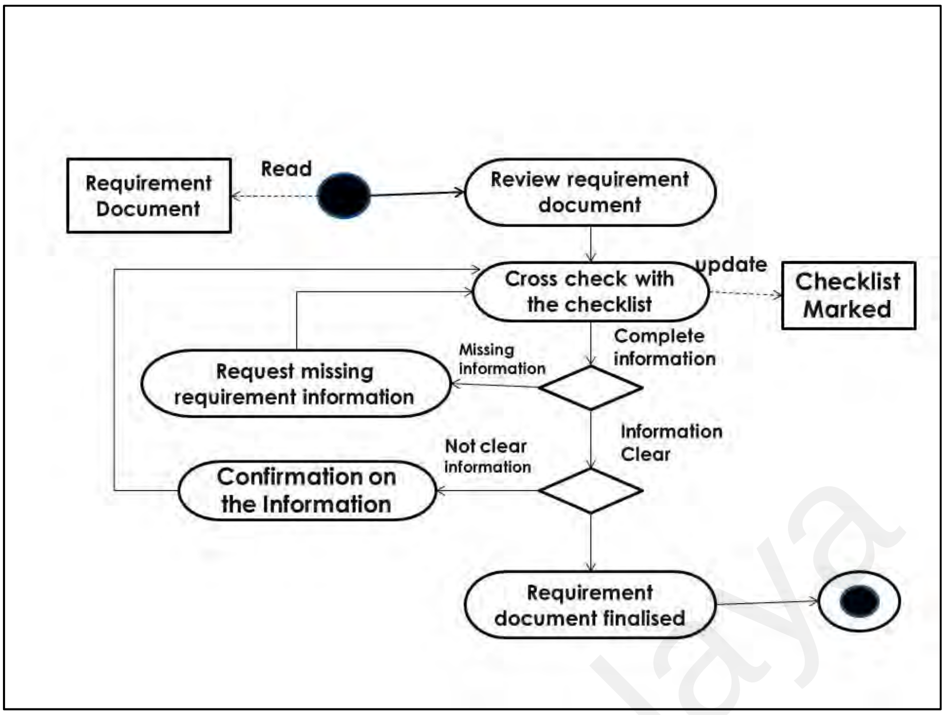
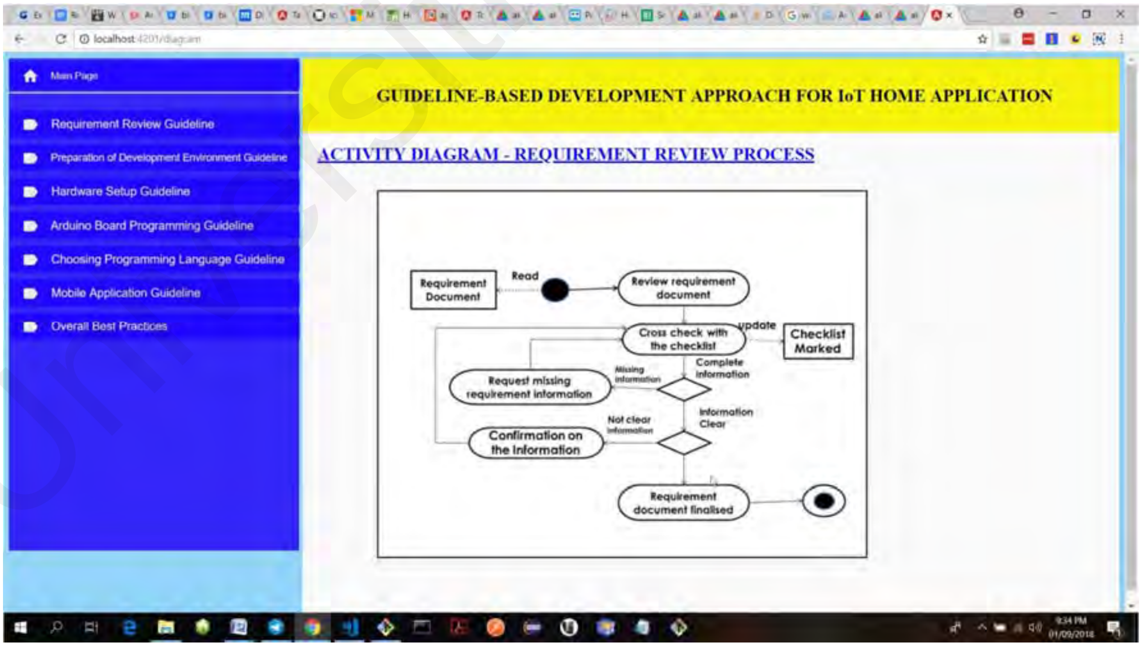

Requirement Document
Read
Review requirement document
Cross check with the checklist
update
Checklist Marked
Missing information
Complete information
Request missing requirement information
Not clear information
Information Clear
Confirmation on the Information
Requirement document finalised

Figure 4.8: The requirements review activity diagram

Universiti Malaya


localhost:4201/diagram
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

ACTIVITY DIAGRAM - REQUIREMENT REVIEW PROCESS

Requirement Document
Read
Review requirement document
Cross check with the checklist
update
Checklist Marked
Missing information
Complete information
Request missing requirement information
Not clear information
Information Clear
Confirmation on the Information
Requirement document finalised

Figure 4.9 : The approach screen on activity diagram

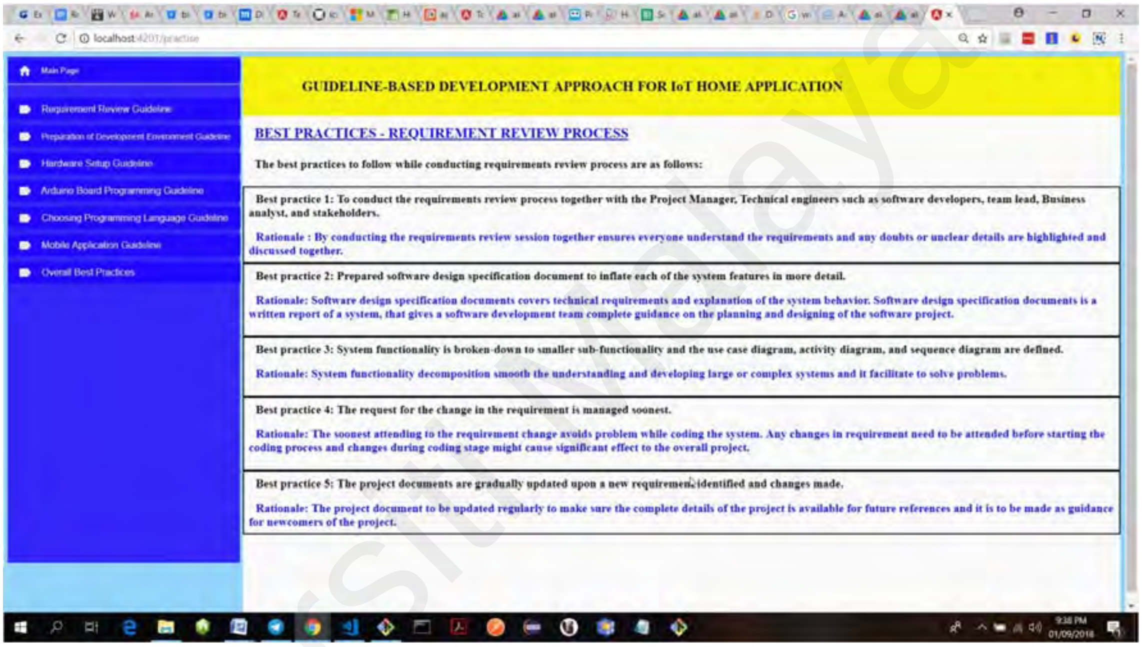
The best practices suggested to follow while conducting requirements review process are as follows:

- Best practice 1: To conduct the requirements review process together with the Project Manager, Technical engineers such as software developers, team lead, Business analyst, and stakeholders. Rationale: By conducting the requirements review session together ensures everyone understand the requirements and any doubts or unclear details are highlighted and discussed together.
- Best practice 2: Prepared software design specification document to inflate each of the system features in more detail. Rationale: Software design specification documents covers technical requirements and explanation of the system behavior. Software design specification documents is a written report of a system, that gives a software development team complete guidance on the planning and designing of the software project
- Best practice 3: System functionality is broken-down to smaller sub-functionality and the use case diagram, activity diagram, and sequence diagram are defined. Rationale: System functionality decomposition smooth the understanding and developing large or complex systems and it facilitate to solve problems.
- Best practice 4: The request for the change in the requirement is managed soonest. Rationale: The soonest attending to the requirement change avoids problem while coding the system. Any changes in requirement need to be attended before starting the coding process and changes during coding stage might cause significant effect to the overall project.
- Best practice 5: The project documents are gradually updated upon a new requirement identified and changes made. Rationale: The project document to be

Universiti Malaya

updated regularly to make sure the complete details of the project is available for future references and it is to be made as guidance for newcomers of the project.

Figure 4.10 shows the approach screen related to best practices of requirements review stage.


localhost:4201/practise
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices
9:38 PM
01/09/2018

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

BEST PRACTICES - REQUIREMENT REVIEW PROCESS

The best practices to follow while conducting requirements review process are as follows:

Best practice 1: To conduct the requirements review process together with the Project Manager, Technical engineers such as software developers, team lead, Business analyst, and stakeholders.

Rationale : By conducting the requirements review session together ensures everyone understand the requirements and any doubts or unclear details are highlighted and discussed together.

Best practice 2: Prepared software design specification document to inflate each of the system features in more detail.

Rationale: Software design specification documents covers technical requirements and explanation of the system behavior. Software design specification documents is a written report of a system, that gives a software development team complete guidance on the planning and designing of the software project.

Best practice 3: System functionality is broken-down to smaller sub-functionality and the use case diagram, activity diagram, and sequence diagram are defined.

Rationale: System functionality decomposition smooth the understanding and developing large or complex systems and it facilitate to solve problems.

Best practice 4: The request for the change in the requirement is managed soonest.

Rationale: The soonest attending to the requirement change avoids problem while coding the system. Any changes in requirement need to be attended before starting the coding process and changes during coding stage might cause significant effect to the overall project.

Best practice 5: The project documents are gradually updated upon a new requirement identified and changes made.

Rationale: The project document to be updated regularly to make sure the complete details of the project is available for future references and it is to be made as guidance for newcomers of the project.

Figure 4.10 : The approach screen on best practices of requirements review process

4.3 Development environment preparation process

Preparation of development environment will be the first step before starting writing code for either the hardware or the mobile application and prior to hardware set up process. The development environment is where the required platforms, process and software tools are prepared to allow the software developers to set up the hardware parts and write their programs for hardware and software application and also testing the developed system. In IoT Home application development, development environment

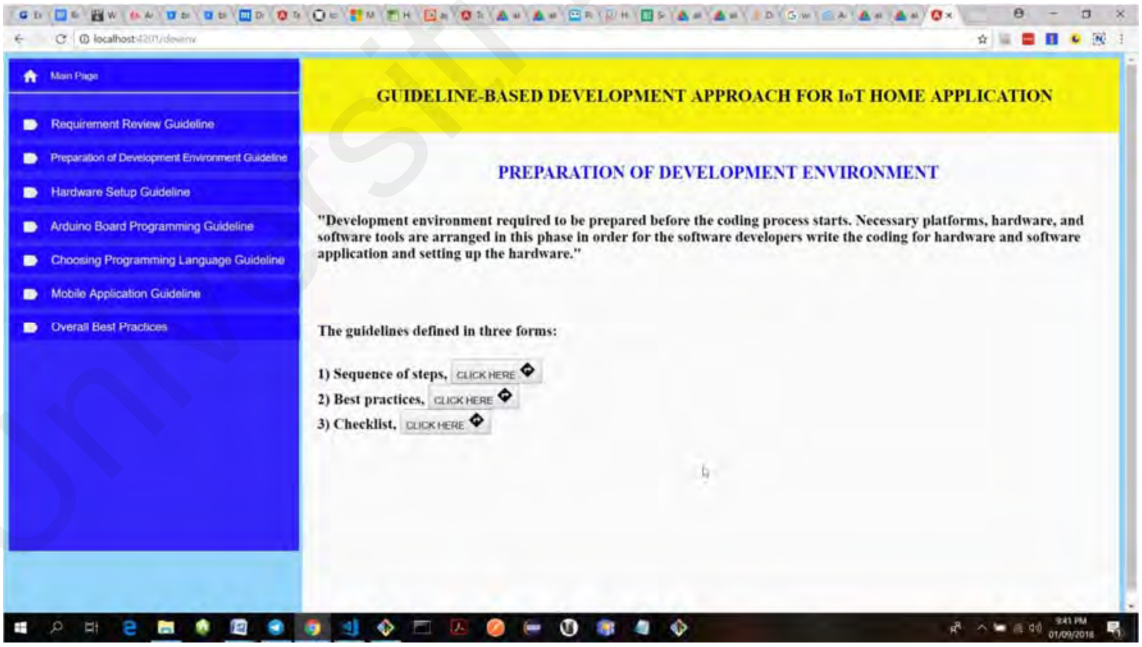
Universiti Malaya

used for: 1) hardware setup covers setting up of microcontroller to decode the IR signal and sending IR signal to smart devices, 2) writing the program to decode as well as sending the IR signal, and 3) writing programs for Android and iOS mobile application. Figure 4.11 illustrate the approach screen of the preparation of development environment process.

- **Guidelines for preparation of development environment**

The guidelines in the form of best practices, checklist and sequence of steps are proposed for the preparation development environment process in the IoT Home application development. The sequence of steps presents the steps involved in this stage. The best practices and checklist subsequently applied throughout the execution of the sequence of steps.

Universiti Malaya


localhost:4201/devenv
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices
CLICK HERE
9:41 PM
01/09/2018

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

PREPARATION OF DEVELOPMENT ENVIRONMENT

"Development environment required to be prepared before the coding process starts. Necessary platforms, hardware, and software tools are arranged in this phase in order for the software developers write the coding for hardware and software application and setting up the hardware."

The guidelines defined in three forms:

1) Sequence of steps, CLICK HERE
2) Best practices, CLICK HERE
3) Checklist, CLICK HERE

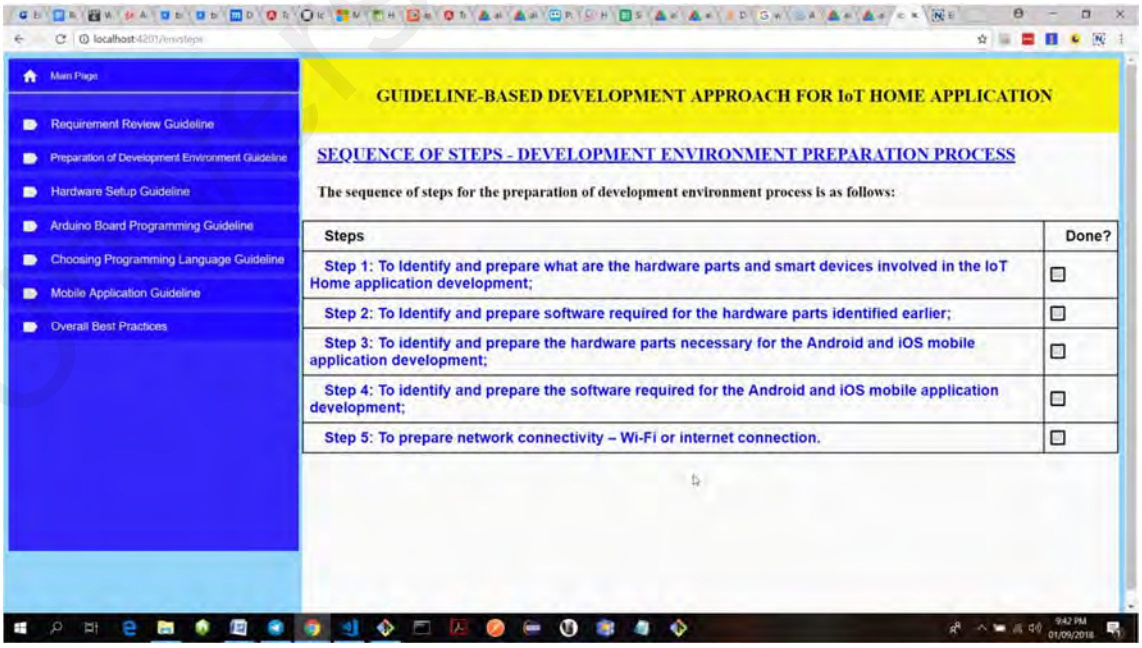
Figure 4.11 : The approach screen of preparation of development environment

There are five steps in the preparation development environment preparation. While conducting these steps, a checklist is important and necessary to assist software developers ensuring a smooth process. Figure 4.12 shows the approach screen related to sequence of steps, Figure 4.13 illustrate the checklist on the approach screen and Figure 4.14 shows the checklist for the preparation of development environment.

The steps involved and proposed for this stage are as follows:

1. To Identify and prepare what are the hardware parts and smart devices involved in the IoT Home application development;
2. To Identify and prepare software required for the hardware parts identified earlier;
3. To identify and prepare the hardware parts necessary for the Android and iOS mobile application development;
4. To identify and prepare the software required for the Android and iOS mobile application development;
5. To prepare network connectivity – Wi-Fi or internet connection.

Universiti Malaya


localhost:4201/envsteps
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices

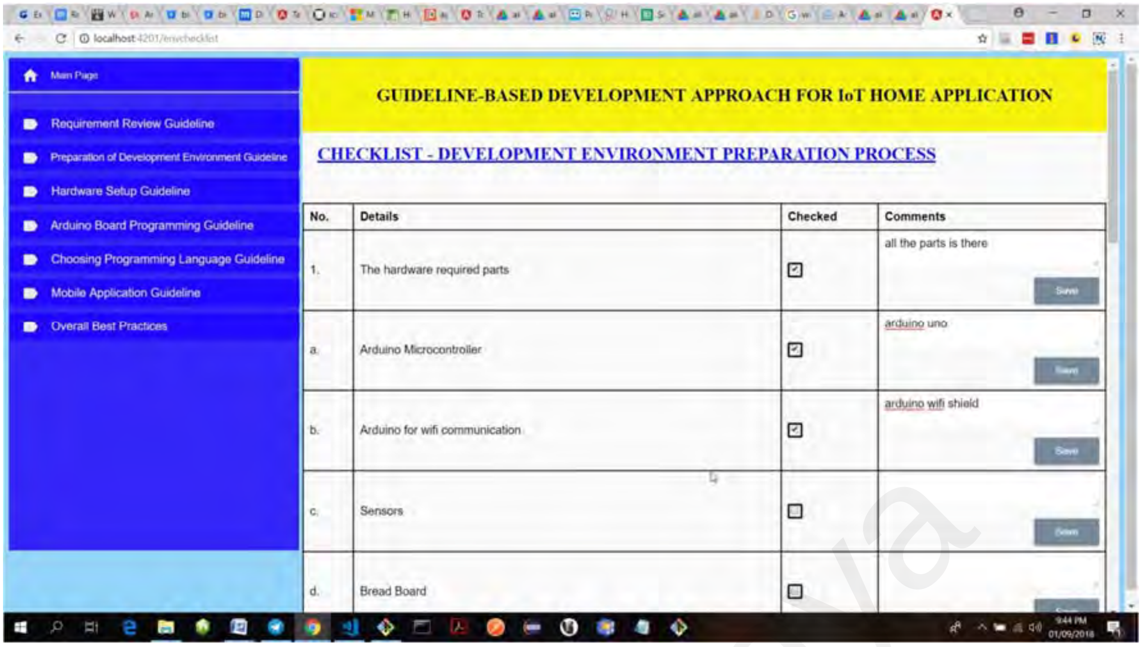
GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

SEQUENCE OF STEPS - DEVELOPMENT ENVIRONMENT PREPARATION PROCESS

The sequence of steps for the preparation of development environment process is as follows:

Steps	Done?
Step 1: To Identify and prepare what are the hardware parts and smart devices involved in the IoT Home application development;	□
Step 2: To Identify and prepare software required for the hardware parts identified earlier;	□
Step 3: To identify and prepare the hardware parts necessary for the Android and iOS mobile application development;	□
Step 4: To identify and prepare the software required for the Android and iOS mobile application development;	□
Step 5: To prepare network connectivity – Wi-Fi or internet connection.	□

Figure 4.12 : The approach screen on sequence of steps of development environment preparation process.


localhost:4201/envchecklist
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices
Save
9:44 PM
01/09/2018

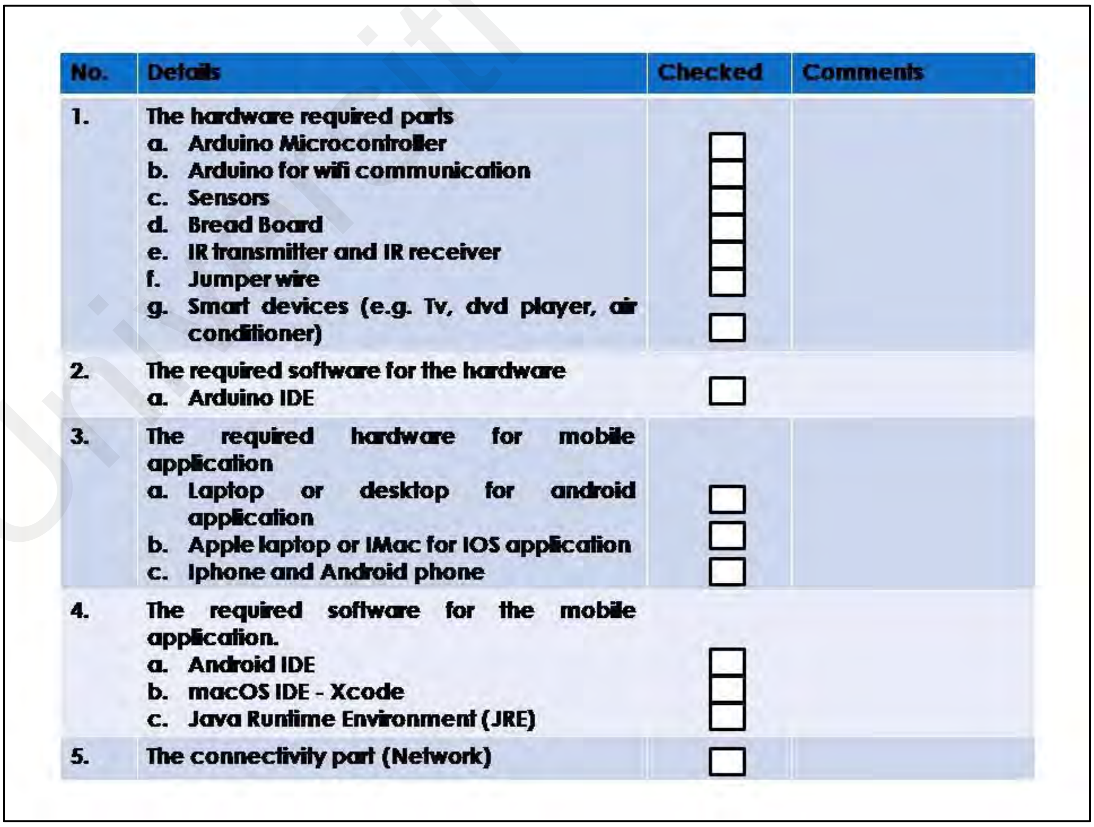
GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

CHECKLIST - DEVELOPMENT ENVIRONMENT PREPARATION PROCESS

No.	Details	Checked	Comments
1.	The hardware required parts	☑	all the parts is there
a.	Arduino Microcontroller	☑	arduino uno
b.	Arduino for wifi communication	☑	arduino wifi shield
c.	Sensors	□	
d.	Bread Board	□	

Figure 4.13 : The approach screen on checklist of development environment preparation process.

Universiti Malaya



No.	Details	Checked	Comments
1.	The hardware required parts a. Arduino Microcontroller b. Arduino for wifi communication c. Sensors d. Bread Board e. IR transmitter and IR receiver f. Jumper wire g. Smart devices (e.g. Tv, dvd player, air conditioner)	□ □ □ □ □ □ □	
2.	The required software for the hardware a. Arduino IDE	□	
3.	The required hardware for mobile application a. Laptop or desktop for android application b. Apple laptop or IMac for IOS application c. Iphone and Android phone	□ □ □	
4.	The required software for the mobile application. a. Android IDE b. macOS IDE - Xcode c. Java Runtime Environment (JRE)	□ □ □	
5.	The connectivity part (Network)	□	

Figure 4.14 : The development environment preparation checklist

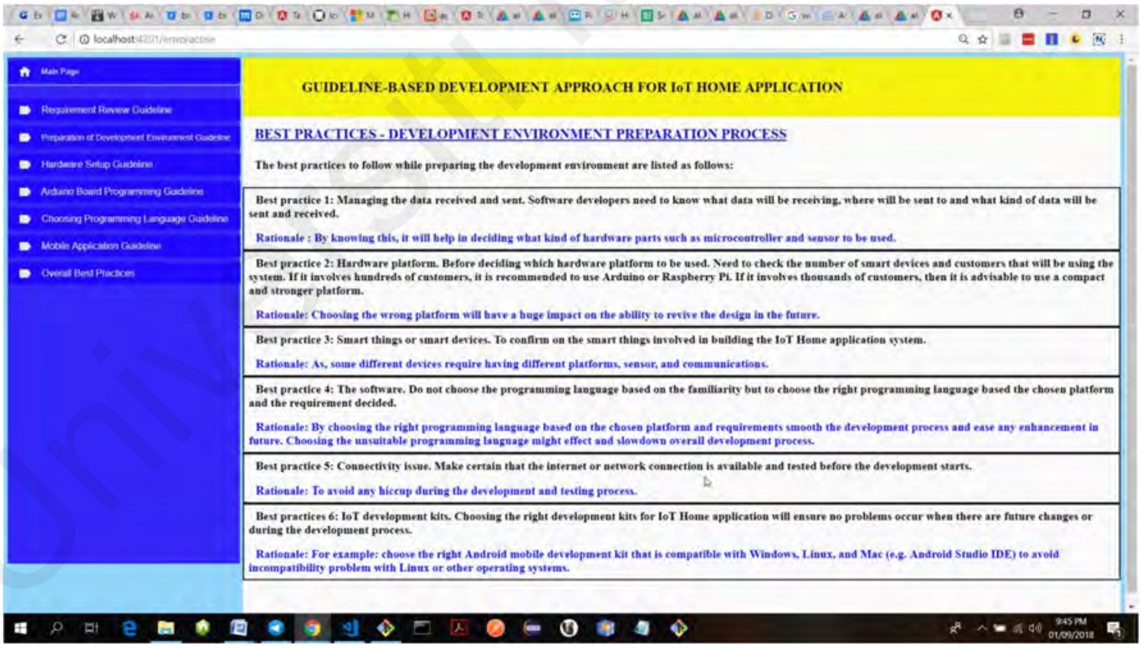
The best practices that are proposed and recommended to follow while preparing the development environment are listed as follows:

- Best practice 1: Managing the data received and sent. Software developers need to know what data will be receiving, where will be sent to and what kind of data will be sent and received. Rationale: By knowing this, it will help in deciding what kind of hardware parts such as microcontroller and sensor to be used.
- Best practice 2: Hardware platform. Before deciding which hardware platform to be used. Need to check the number of smart devices and customers that will be using the system. If it involves hundreds of customers, it is recommended to use Arduino or Raspberry Pi. If it involves thousands of customers, then it is advisable to use a compact and stronger platform. Rationale: Choosing the wrong platform will have a huge impact on the ability to revive the design in the future.
- Best practice 3: Smart things or smart devices. To confirm on the smart things involved in building the IoT Home application system. Rationale: As, some different devices require having different platforms, sensor, and communications.
- Best practice 4: The software. Do not choose the programming language based on the familiarity but to choose the right programming language based the chosen platform and the requirement decided. Rationale: By choosing the right programming language based on the chosen platform and requirements smooth the development process and ease any enhancement in future. Choosing the unsuitable programming language might effect and slowdown overall development process.

Universiti Malaya

- Best practice 5: Connectivity issue. Make certain that the internet or network connection is available and tested before the development starts. Rationale: to avoid any hiccup during the development and testing process.
- Best practices 6: IoT development kits. Choosing the right development kits for IoT Home application will ensure no problems occur when there are future changes or during the development process. Rationale: For example: choose the right Android mobile development kit that is compatible with Windows, Linux, and Mac (e.g. Android Studio IDE) to avoid incompatibility problem with Linux or other operating systems.

Figure 4.15 shows the best practices approach screen on the preparation of development environment.


localhost:4201/envpractise
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices
9:45 PM
01/09/2018

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

BEST PRACTICES - DEVELOPMENT ENVIRONMENT PREPARATION PROCESS

The best practices to follow while preparing the development environment are listed as follows:

Best practice 1: Managing the data received and sent. Software developers need to know what data will be receiving, where will be sent to and what kind of data will be sent and received.

Rationale : By knowing this, it will help in deciding what kind of hardware parts such as microcontroller and sensor to be used.

Best practice 2: Hardware platform. Before deciding which hardware platform to be used. Need to check the number of smart devices and customers that will be using the system. If it involves hundreds of customers, it is recommended to use Arduino or Raspberry Pi. If it involves thousands of customers, then it is advisable to use a compact and stronger platform.

Rationale: Choosing the wrong platform will have a huge impact on the ability to revive the design in the future.

Best practice 3: Smart things or smart devices. To confirm on the smart things involved in building the IoT Home application system.

Rationale: As, some different devices require having different platforms, sensor, and communications.

Best practice 4: The software. Do not choose the programming language based on the familiarity but to choose the right programming language based the chosen platform and the requirement decided.

Rationale: By choosing the right programming language based on the chosen platform and requirements smooth the development process and ease any enhancement in future. Choosing the unsuitable programming language might effect and slowdown overall development process.

Best practice 5: Connectivity issue. Make certain that the internet or network connection is available and tested before the development starts.

Rationale: To avoid any hiccup during the development and testing process.

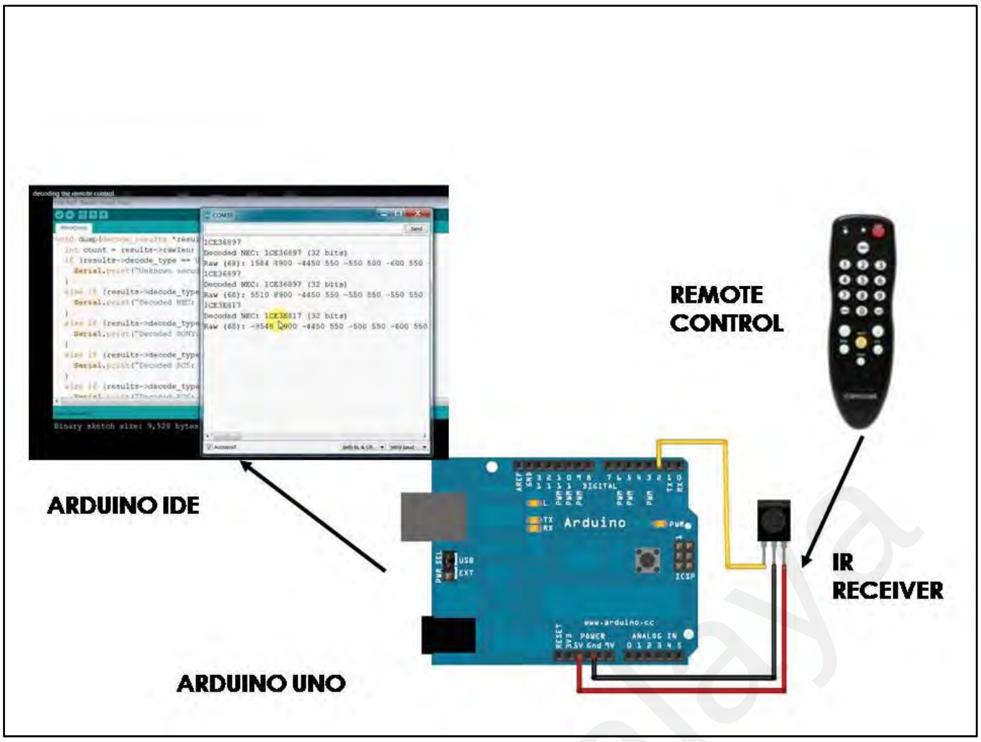
Best practices 6: IoT development kits. Choosing the right development kits for IoT Home application will ensure no problems occur when there are future changes or during the development process.

Rationale: For example: choose the right Android mobile development kit that is compatible with Windows, Linux, and Mac (e.g. Android Studio IDE) to avoid incompatibility problem with Linux or other operating systems.

Figure 4.15 : The approach screen on best practices related to development environment preparation process

Example of hardware and software required for the hardware development part is shown in Figure 4.16 and Figure 4.17.

Universiti Malaya



```
1CE36897
Decoded NEC: 1CE36897 (32 bits)
Raw (68): 1584 8900 -4450 550 -550 500 -600 550
1CE36897
Decoded NEC: 1CE36897 (32 bits)
Raw (68): 5510 8900 -4450 550 -550 550 -550 550
1CE3E817
Decoded NEC: 1CE3E817 (32 bits)
Raw (68): -9548 8900 -4450 550 -500 550 -600 550
```

Figure 4.16 : The hardware and software required for the decoding process of the hardware setup part

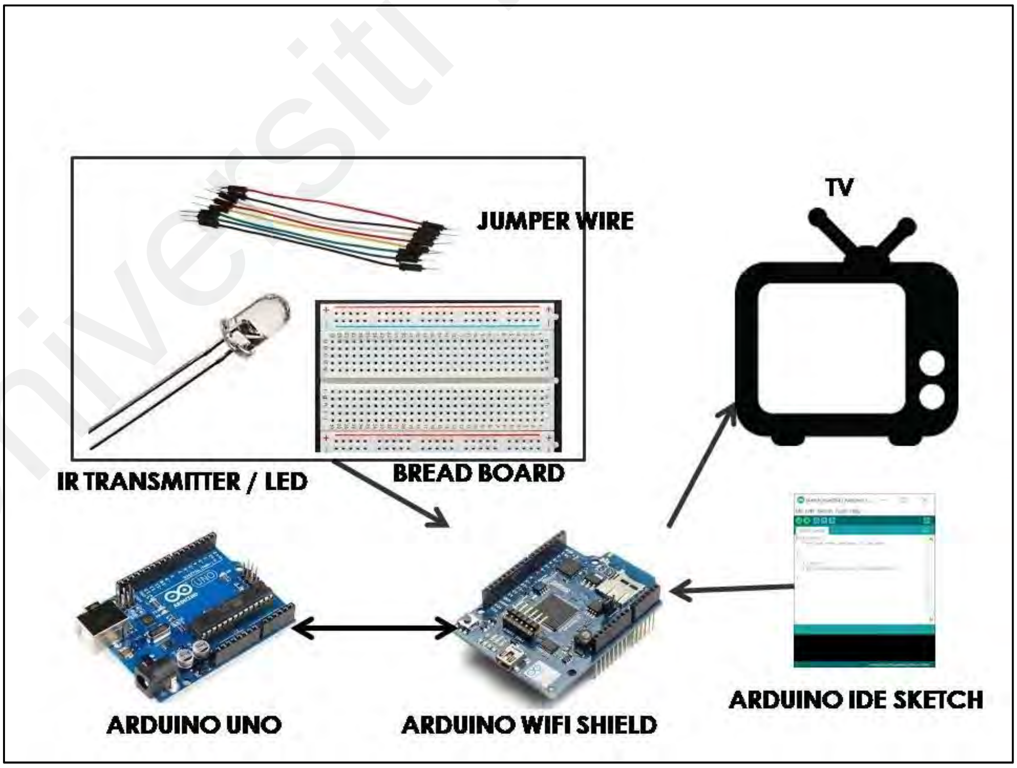


Figure 4.17 : The hardware and software required for the transmitting IR signal process of the hardware setup part

Universiti Malaya

4.4 Hardware setup process for IoT Home application

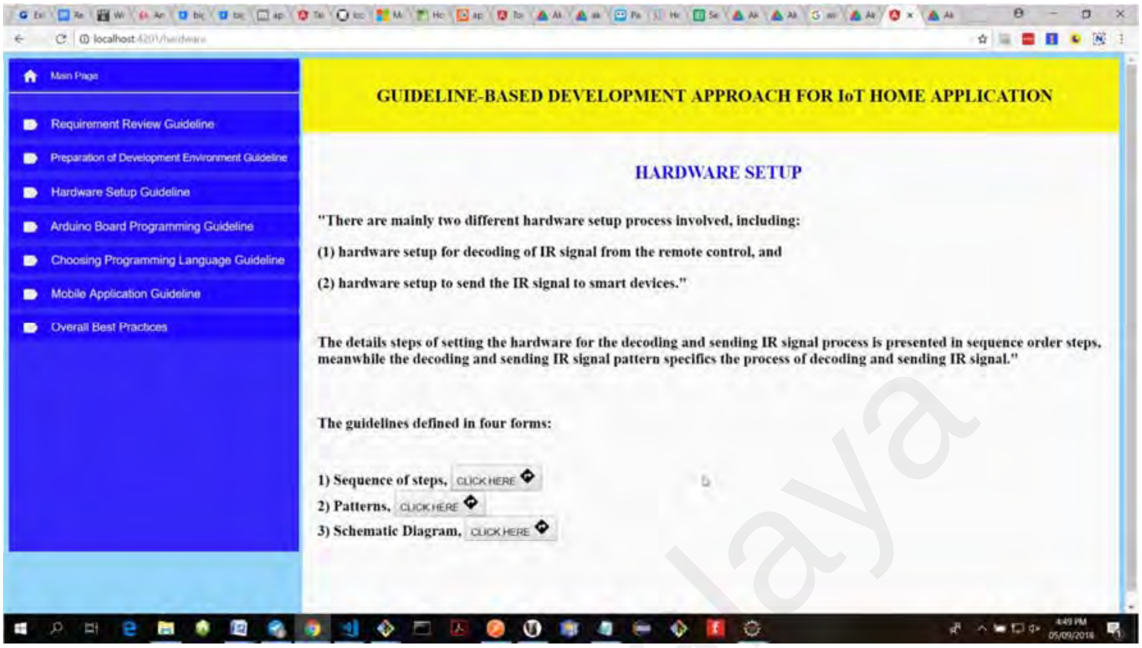
Hardware setup process for IoT Home application is divided into two different hardware setup processes. In this research, the chosen platform is Arduino. The reasons in choosing Arduino are:

- Arduino is an open source platform for electronics prototyping grounded on adaptable and easy to use hardware and software.
- Arduino opens the whole hardware designs and interconnected data to all, enables users unreservedly plan, design and execute their ideas.
- Arduino consists of single board microcontroller with multiple input and output support.
- Arduino able to sense surroundings by receiving various sensors inputs and affects environments via controlling other actuators, motors, and lights. Besides, users able exchange information to any devices with network.
- Firmware runs in Arduino permits Arduino to interconnect with any computer using USB and allows access to work and upload programs to Arduino.

The above listed reason and as can be seen from the previous study conducted, the researchers have focused on Arduino as the platform for the IoT Home application system development which have directed in choosing Arduino for IoT home application development. Besides, Arduino is suitable for projects that involve sensors to read and send data.

The first hardware setup will be to receive and demodulate the IR signal received from the remote control of any home appliances. Meanwhile, the second hardware setup is to transmit the IR codes to home appliances such as TV, DVD player, and air conditioners. Figure 4.18 shows the approach screen on the hardware setup main page.

Universiti Malaya



Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices
localhost:4201/hardware

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

HARDWARE SETUP

"There are mainly two different hardware setup process involved, including:

(1) hardware setup for decoding of IR signal from the remote control, and

(2) hardware setup to send the IR signal to smart devices."

The details steps of setting the hardware for the decoding and sending IR signal process is presented in sequence order steps, meanwhile the decoding and sending IR signal pattern specifics the process of decoding and sending IR signal."

The guidelines defined in four forms:

1) Sequence of steps, CLICK HERE

2) Patterns, CLICK HERE

3) Schematic Diagram, CLICK HERE

Figure 4.18 : The approach screen on the hardware setup main page

Universiti Malaya

- **Guidelines for Hardware set up**

The guidelines will be in the forms of: sequence of steps and pattern of the hardware setup for the two different processes. The details steps of setting the hardware for the decoding and sending IR signal process is presented in sequence of steps, meanwhile the decoding and sending IR signal pattern specifics the process of decoding and sending IR signal. The approach screen on the sequence of steps of hardware setup is shown in Figure 4.19 and the approach screen on the schematic diagram for setup hardware on decoding IR signal process and sending IR signal is shown in Figure 4.20.

1. Hardware setup to receive and decode the IR signals from the remote control
   a. Wire IR receiver (e.g. IR photo sensor) to Arduino UNO – Pin 1 to pin 11 at Arduino Uno, Pin 2 to Ground at Arduino Uno, and Pin 3 to GV at Arduino Uno. (Jumper wire and breadboard is used for the wiring part).

Figure 4.21 shows the schematic diagram for setup of hardware regards to receive and decode the remote control's IR signals.

b. Download IR library at the Arduino IDE (Arduino IDE already installed in the development environment process in the previous chapter).

c. Find IRrecvDump file from the Arduino library, then modify it and upload the sketch to encode the signals from the remote control.

d. Open serial monitor and press the remote control button by pointing to the IR photo sensor. The respective code related to the value of the button pressed will appear at the serial monitor screen. Take note of the codes (e.g. PANASONIC: 4004 100BCBD (48 bits)).

2. Hardware setup to send the IR signals to home appliances

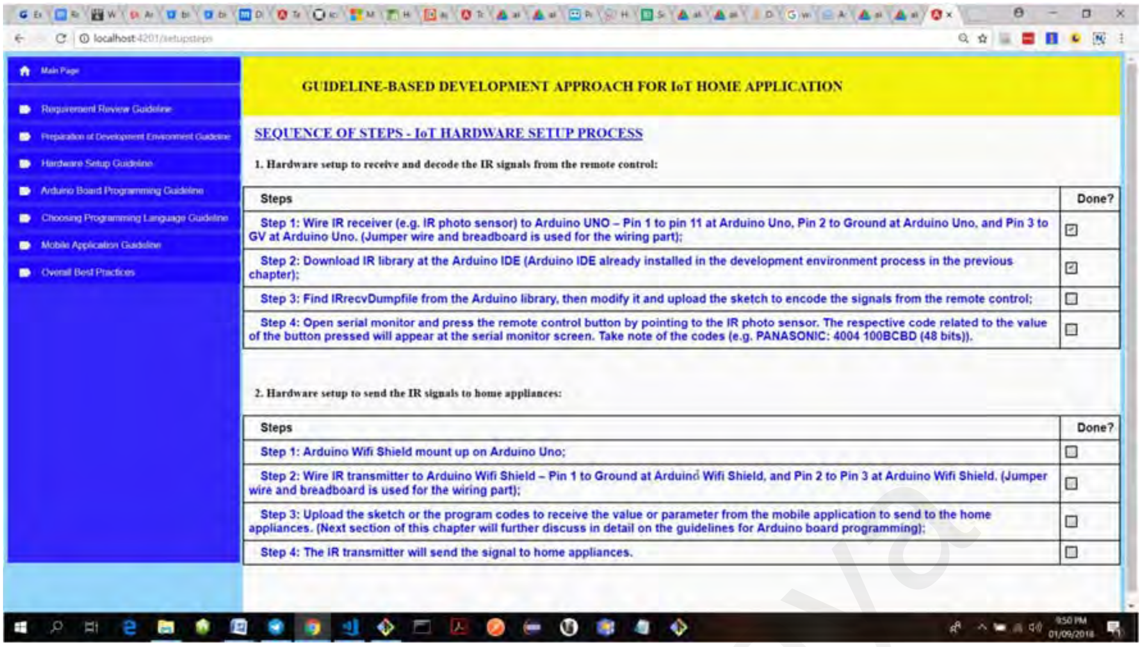
   a. Arduino Wifi Shield mount up on Arduino Uno

   b. Wire IR transmitter to Arduino Wifi Shield – Pin 1 to Ground at Arduino Wifi Shield, and Pin 2 to Pin 3 at Arduino Wifi Shield. (Jumper wire and breadboard is used for the wiring part). Figure 4.22 shows the schematic diagram for setup of hardware regards to sending IR signals to home devices.

   c. Upload the sketch or the program codes to receive the value or parameter from the mobile application to send to the home appliances. (Next section of this chapter will further discuss in detail on the guidelines for Arduino board programming).

   d. The IR transmitter will send the signal to home appliances.

Universiti Malaya


localhost:4201/setupsteps
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

SEQUENCE OF STEPS - IoT HARDWARE SETUP PROCESS

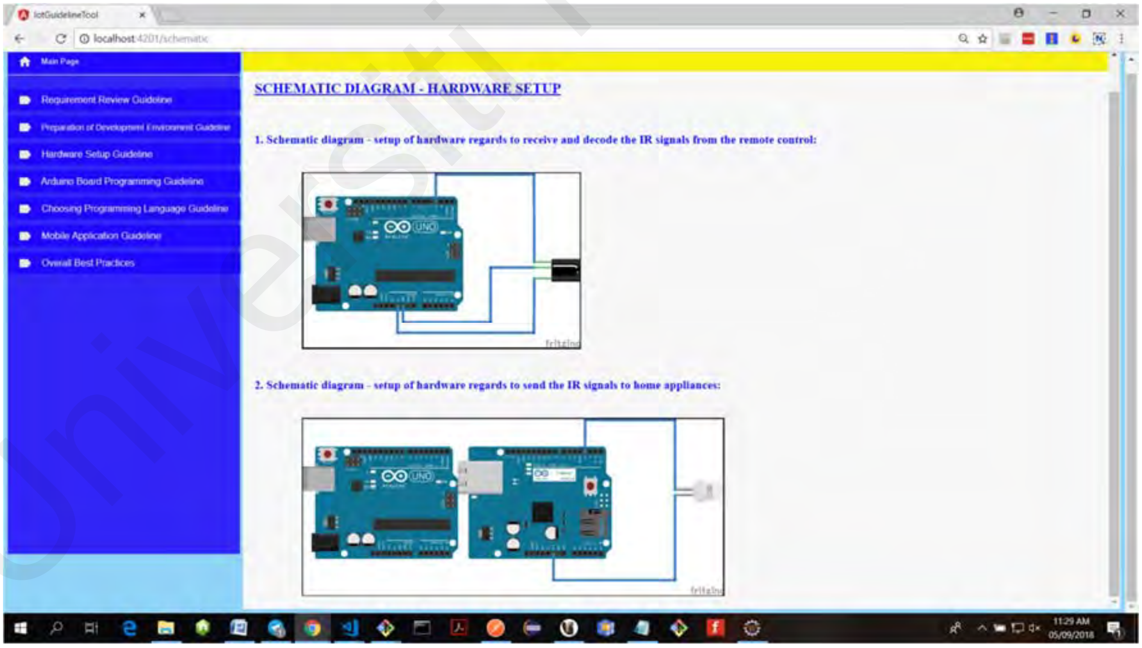
1. Hardware setup to receive and decode the IR signals from the remote control:

Steps	Done?
Step 1: Wire IR receiver (e.g. IR photo sensor) to Arduino UNO – Pin 1 to pin 11 at Arduino Uno, Pin 2 to Ground at Arduino Uno, and Pin 3 to GV at Arduino Uno. (Jumper wire and breadboard is used for the wiring part);	☑
Step 2: Download IR library at the Arduino IDE (Arduino IDE already installed in the development environment process in the previous chapter);	☑
Step 3: Find IRrecvDumpfile from the Arduino library, then modify it and upload the sketch to encode the signals from the remote control;	□
Step 4: Open serial monitor and press the remote control button by pointing to the IR photo sensor. The respective code related to the value of the button pressed will appear at the serial monitor screen. Take note of the codes (e.g. PANASONIC: 4004 100BCBD (48 bits)).	□

2. Hardware setup to send the IR signals to home appliances:

Steps	Done?
Step 1: Arduino Wifi Shield mount up on Arduino Uno;	□
Step 2: Wire IR transmitter to Arduino Wifi Shield – Pin 1 to Ground at Arduind Wifi Shield, and Pin 2 to Pin 3 at Arduino Wifi Shield. (Jumper wire and breadboard is used for the wiring part);	□
Step 3: Upload the sketch or the program codes to receive the value or parameter from the mobile application to send to the home appliances. (Next section of this chapter will further discuss in detail on the guidelines for Arduino board programming);	□
Step 4: The IR transmitter will send the signal to home appliances.	□

Figure 4.19 : The approach screen on sequence of steps of hardware setup.


IotGuidelineTool
localhost:4201/schematic
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices
UNO
fritzing
UNO
fritzing

SCHEMATIC DIAGRAM - HARDWARE SETUP

1. Schematic diagram - setup of hardware regards to receive and decode the IR signals from the remote control:

2. Schematic diagram - setup of hardware regards to send the IR signals to home appliances:

Figure 4.20 : The approach screen on the schematic diagram of hardware setup

Universiti Malaya

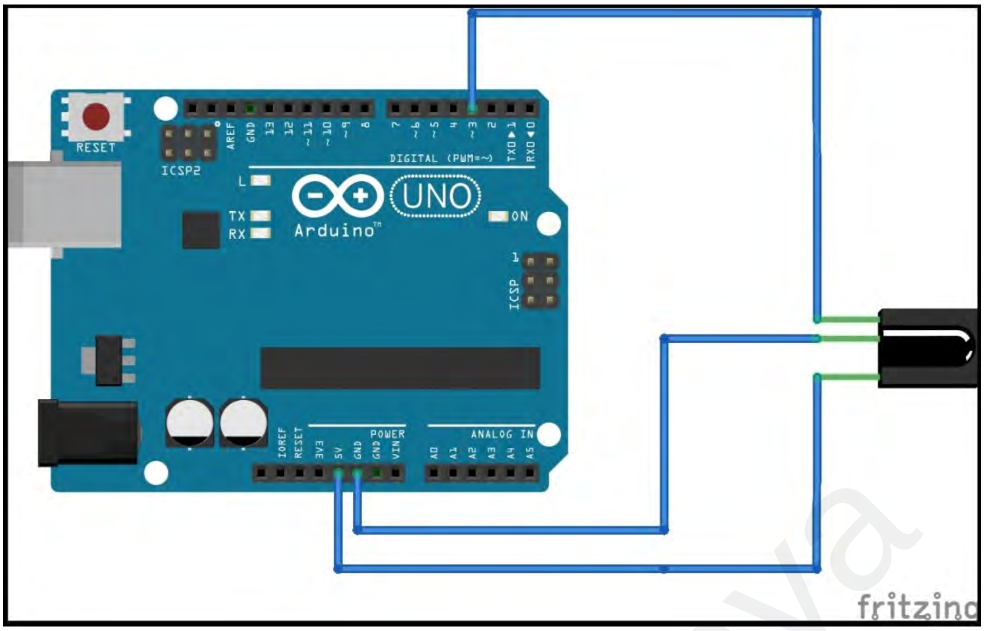


Figure 4.21 : The schematic diagram on the hardware setup for decoding IR signal

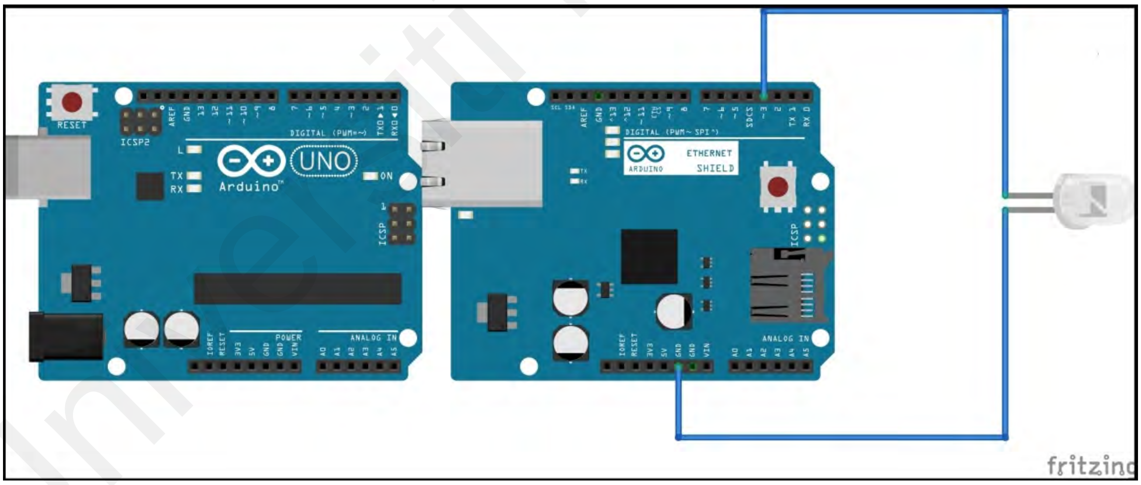


Figure 4.22 : The schematic diagram on hardware setup for sending IR signal to home appliances

The patterns for the above two hardware setup processes are illustrated in Figure 4.23 and Figure 4.25. Meanwhile, the approach screen related to the decoding ir-

Universiti Malaya

receiver and transmitting ir-receiver pattern of the hardware setup is shown in Figure 4.24 and Figure 4.26.

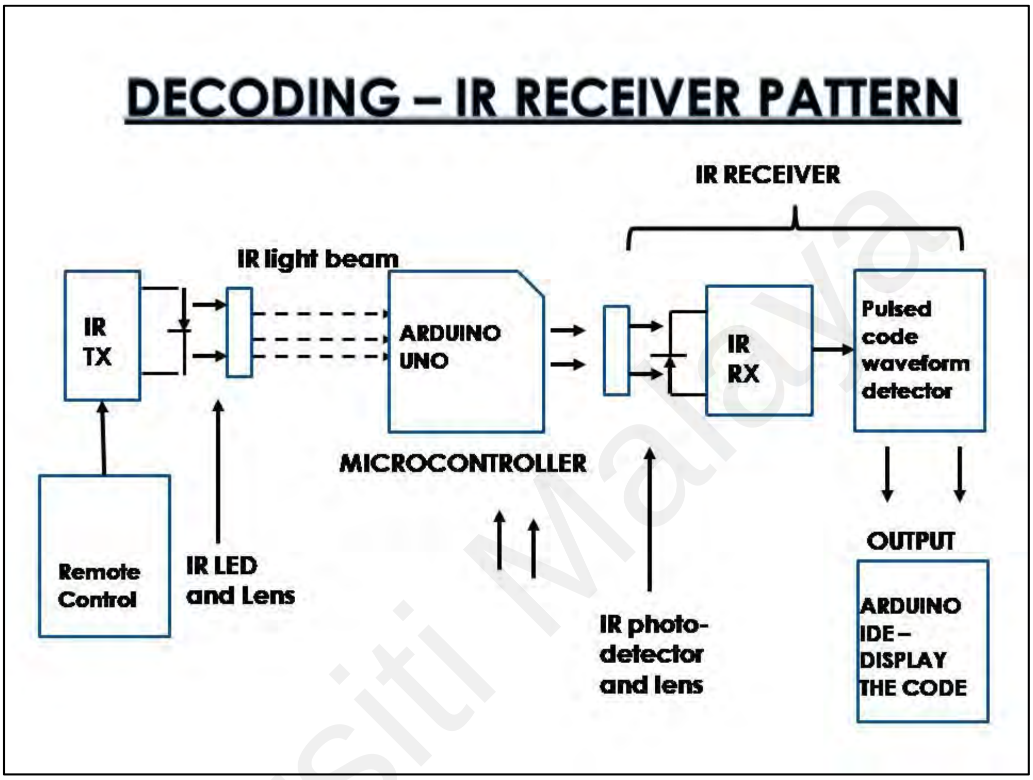
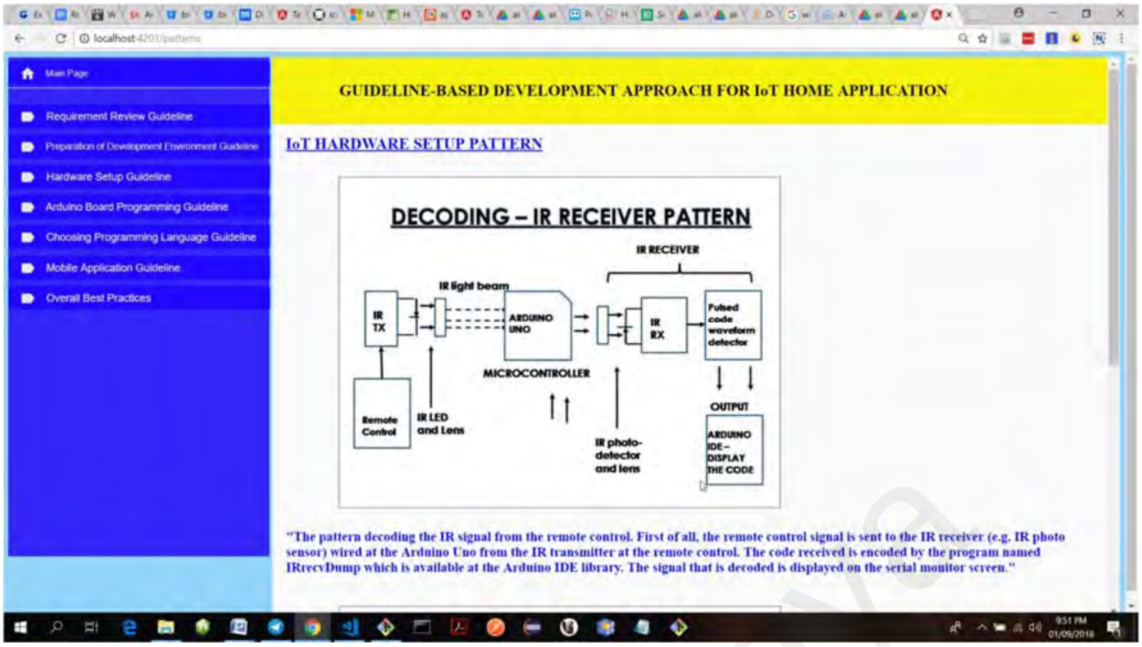

DECODING – IR RECEIVER PATTERN
IR RECEIVER
IR light beam
IR TX
ARDUINO UNO
IR RX
Pulsed code waveform detector
MICROCONTROLLER
OUTPUT
Remote Control
IR LED and Lens
IR photo-detector and lens
ARDUINO IDE – DISPLAY THE CODE

Figure 4.23 : Pattern of decoding process

Figure 4.23 shows the pattern decoding the IR signal from the remote control. First of all, the remote control signal is sent to the IR receiver (e.g. IR photo sensor) wired at the Arduino Uno from the IR transmitter at the remote control. The code received is encoded by the program named IRrecvDump which is available at the Arduino IDE library. The signal that is decoded is displayed on the serial monitor screen.

Universiti Malaya


localhost:4201/patterns
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices
DECODING – IR RECEIVER PATTERN
IR RECEIVER
IR light beam
IR TX
ARDUINO UNO
IR RX
Pulsed code waveform detector
MICROCONTROLLER
OUTPUT
Remote Control
IR LED and Lens
IR photo-detector and lens
ARDUINO IDE – DISPLAY THE CODE
9:51 PM
01/09/2018

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

IoT HARDWARE SETUP PATTERN

"The pattern decoding the IR signal from the remote control. First of all, the remote control signal is sent to the IR receiver (e.g. IR photo sensor) wired at the Arduino Uno from the IR transmitter at the remote control. The code received is encoded by the program named IRrecvDump which is available at the Arduino IDE library. The signal that is decoded is displayed on the serial monitor screen."

Figure 4.24 : The approach screen on Decoding-IR receiver pattern of hardware setup

Universiti Malaya

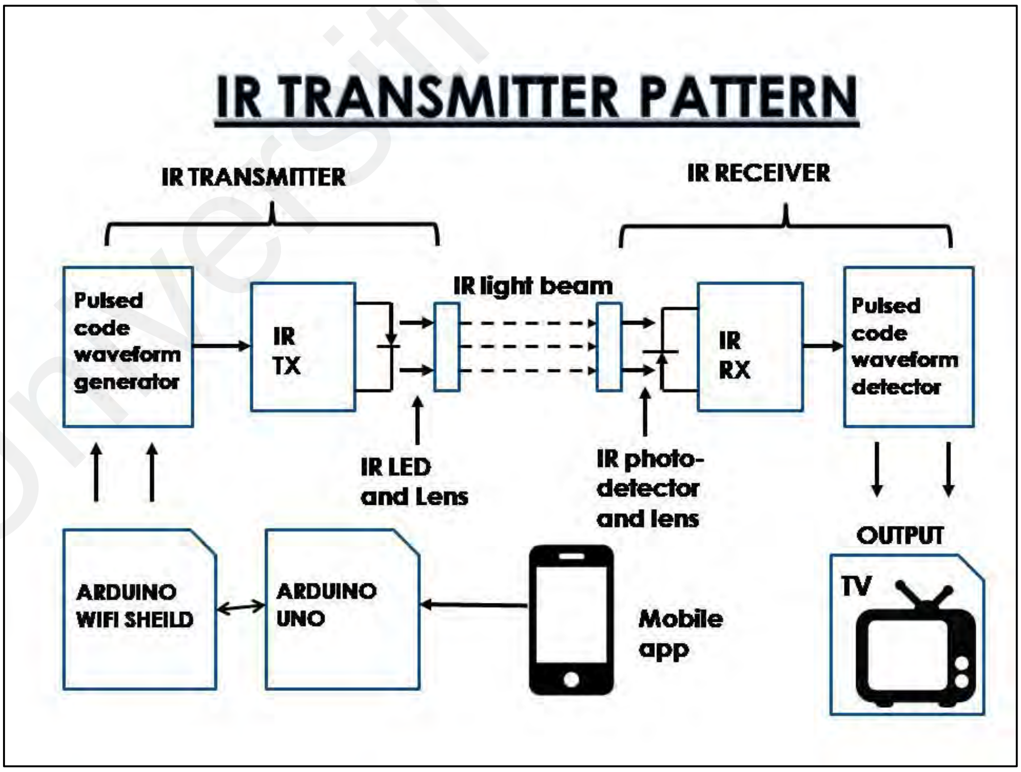
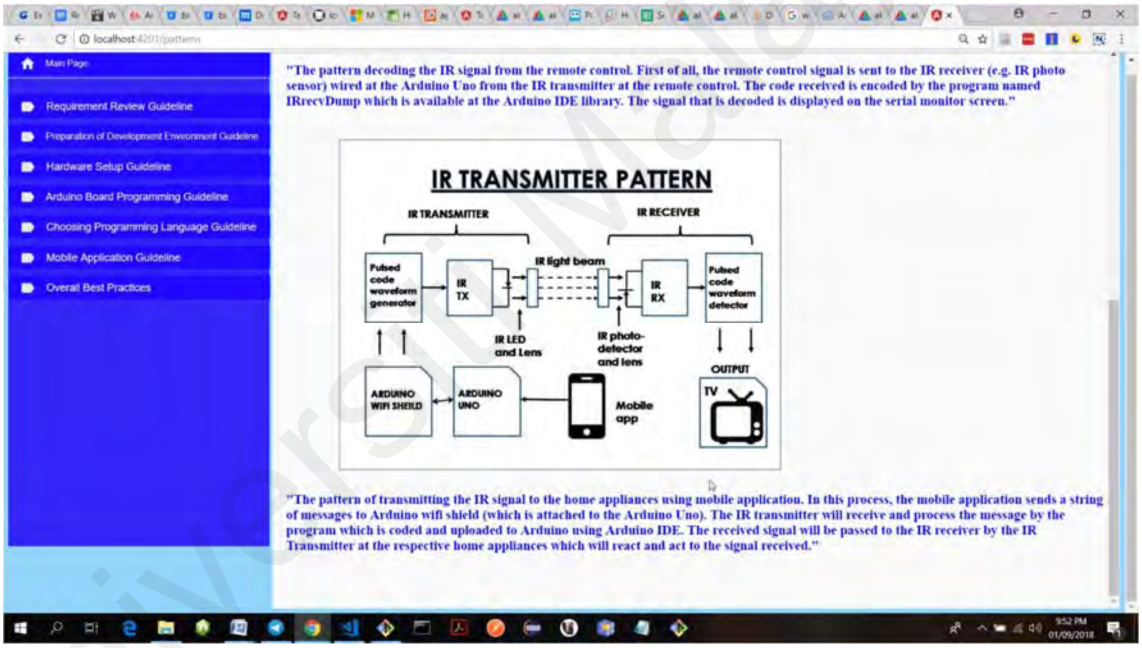


Figure 4.25 : Pattern of IR transmitting process

Figure 4.25 shows the pattern of transmitting the IR signal to the home appliances using mobile application. In this process, the mobile application sends a string of messages to Arduino wifi shield (which is attached to the Arduino Uno). The IR transmitter will receive and process the message by the program which is coded and uploaded to Arduino using Arduino IDE. The received signal will be passed to the IR receiver by the IR Transmitter at the respective home appliances which will react and act to the signal received.


localhost:4201/patterns
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices
IR TRANSMITTER PATTERN
IR TRANSMITTER
IR RECEIVER
Pulsed code waveform generator
IR TX
IR light beam
IR RX
Pulsed code waveform detector
IR LED and Lens
IR photo-detector and lens
OUTPUT
ARDUINO WIFI SHEILD
ARDUINO UNO
Mobile app
TV
9:52 PM
01/09/2018

"The pattern decoding the IR signal from the remote control. First of all, the remote control signal is sent to the IR receiver (e.g. IR photo sensor) wired at the Arduino Uno from the IR transmitter at the remote control. The code received is encoded by the program named IRrecvDump which is available at the Arduino IDE library. The signal that is decoded is displayed on the serial monitor screen."

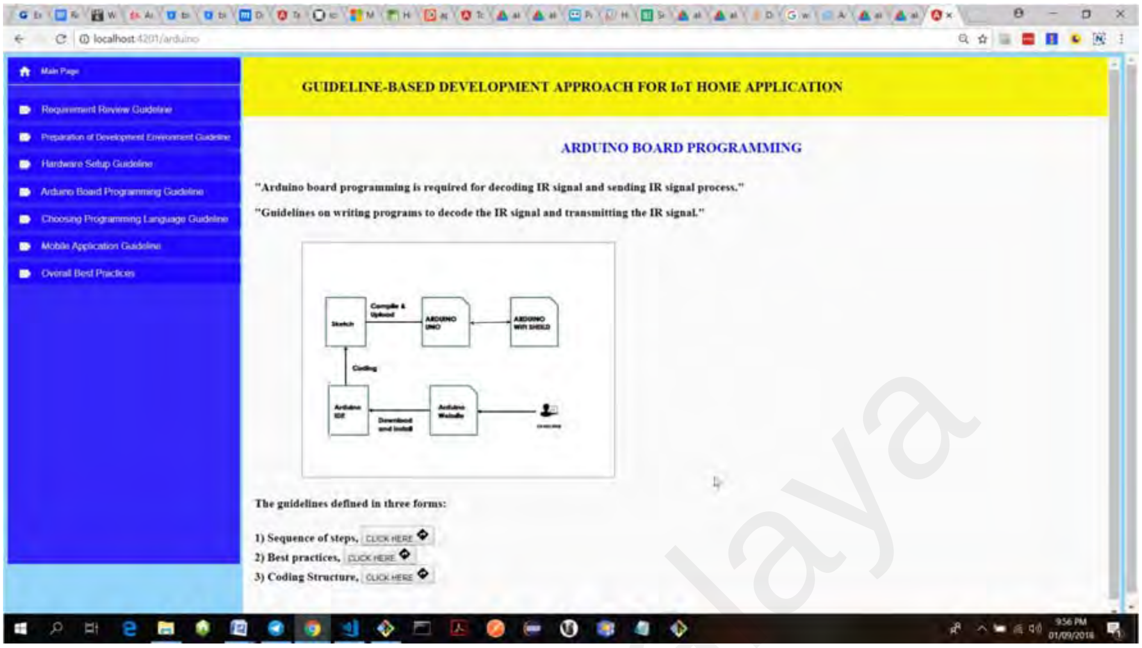
"The pattern of transmitting the IR signal to the home appliances using mobile application. In this process, the mobile application sends a string of messages to Arduino wifi shield (which is attached to the Arduino Uno). The IR transmitter will receive and process the message by the program which is coded and uploaded to Arduino using Arduino IDE. The received signal will be passed to the IR receiver by the IR Transmitter at the respective home appliances which will react and act to the signal received."

Figure 4.26 : The approach screen on IR transmitter pattern of hardware setup

4.5 Arduino board programming process

As mentioned in the earlier section, the Arduino decoding and transmitting process requires theprogram to be coded. In this section, we will propose the guidelines on writing programs to decode the IR signal and transmitting the IR signal. Figure 4.27 shows the approach screen on the Arduino board programming main page.

Universiti Malaya



GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

ARDUINO BOARD PROGRAMMING

"Arduino board programming is required for decoding IR signal and sending IR signal process."

"Guidelines on writing programs to decode the IR signal and transmitting the IR signal."

The guidelines defined in three forms:

1) Sequence of steps,
2) Best practices,
3) Coding Structure,

Figure 4.27 : The approach screen on Arduino board programming process main page

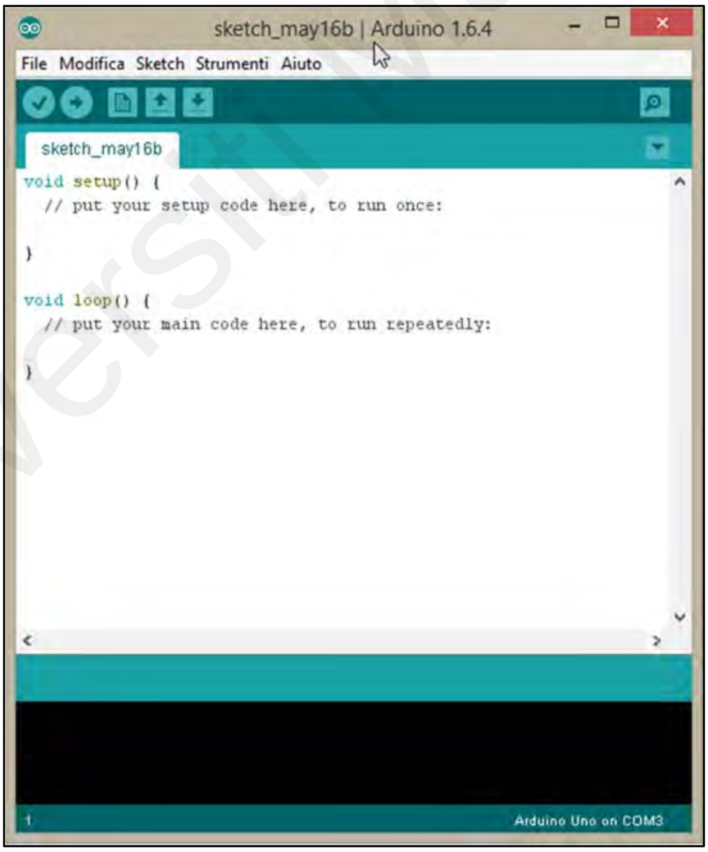
- **Guidelines for Arduino board programming process**

The guidelines will be in the forms of: 1) basic steps, and 2) best practices and 3) coding structure on writing the programs for decoding and transmitting the IR signal. The best practices to be adapted while following the basic steps in Arduino board programming process. The coding structure is to provide a basic knowledge of the Arduino sketch to smooth the programming process. There are basic steps that need to be followed before start writing code for the programs. Figure 4.29 shows the flow pattern of the Arduino board programming process and Figure 4.30 shows the approach screen on the sequence of steps. The basic steps are as follows:

Universiti Malaya

1. Download and Install Arduino IDE. The Arduino software package based on the operating system can be downloaded from https://www.arduino.cc/en/Main/Software.
2. Initial environment setup. At the tool menu, select the board and select the type of the Arduino used such as Arduino Uno.
3. Start writing the code. C++ language is used to write the code. The written code is known as sketches. Figure 4.28 shows the new Arduino sketch page to write the code.
4. Compile the written code and upload it.

Universiti Malaya


sketch_may16b | Arduino 1.6.4
File Modifica Sketch Strumenti Aiuto
sketch_may16b
1
Arduino Uno on COM3

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Figure 4.28 : The Arduino IDE new sketch page

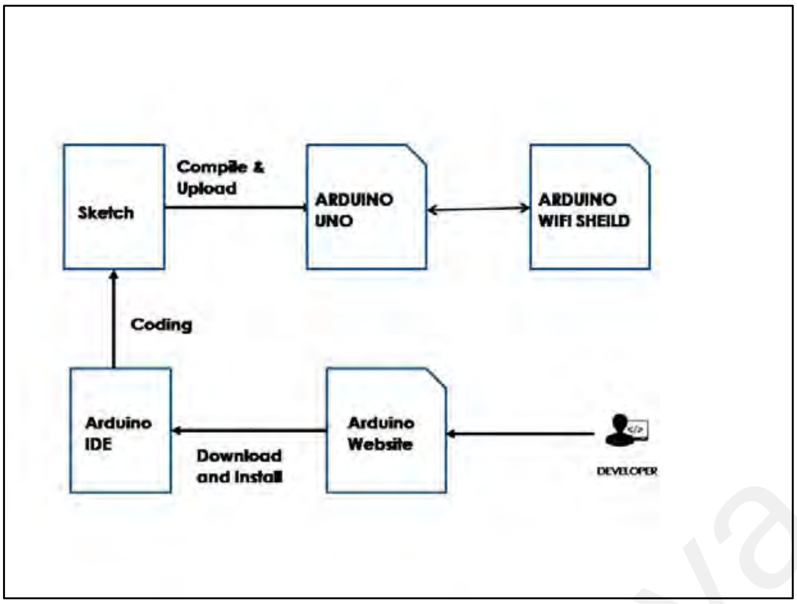
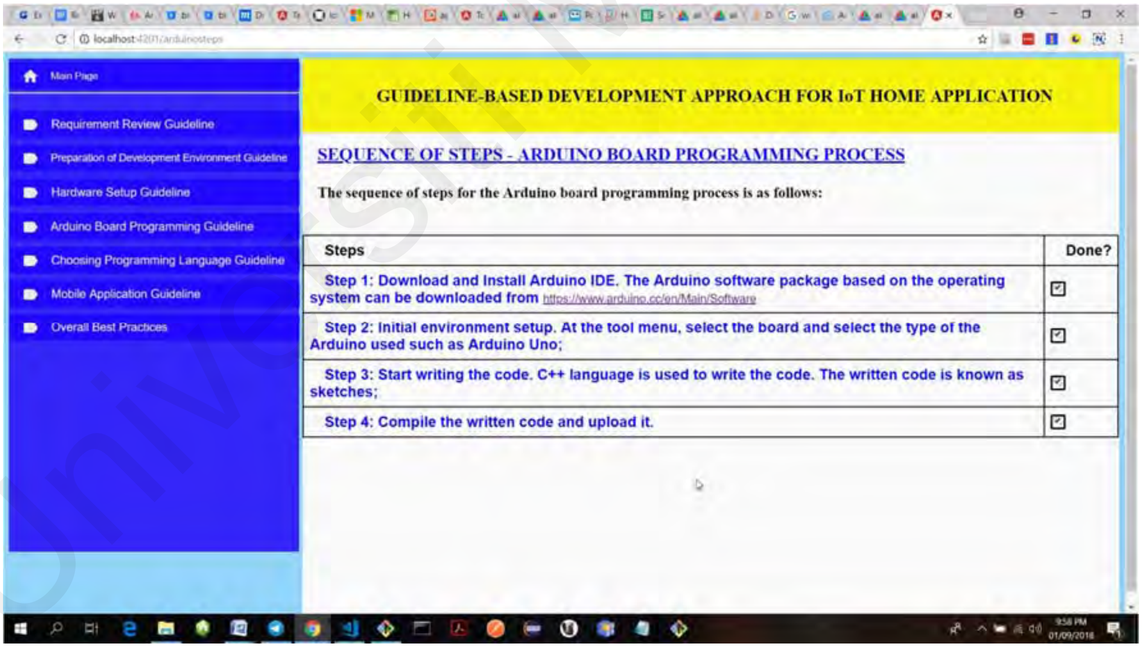


Figure 4.29 : The Arduino board programming flow pattern


localhost:4201/arduinosteps
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

SEQUENCE OF STEPS - ARDUINO BOARD PROGRAMMING PROCESS

The sequence of steps for the Arduino board programming process is as follows:

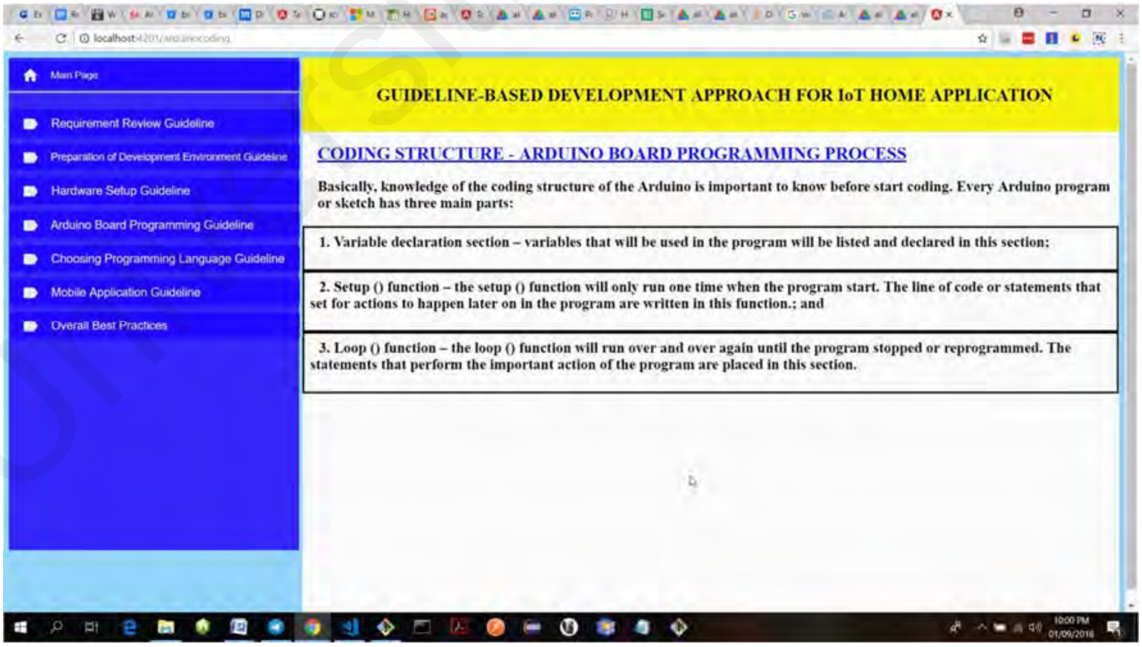
Steps	Done?
Step 1: Download and Install Arduino IDE. The Arduino software package based on the operating system can be downloaded from https://www.arduino.cc/en/Main/Software	☑
Step 2: Initial environment setup. At the tool menu, select the board and select the type of the Arduino used such as Arduino Uno;	☑
Step 3: Start writing the code. C++ language is used to write the code. The written code is known as sketches;	☑
Step 4: Compile the written code and upload it.	☑

Figure 4.30 : The approach screen on sequence of steps of Arduino board programming

Universiti Malaya

Basically, knowledge of the coding structure of the Arduino is important to know before start coding. The coding structure of Arduino board programming approach screen is shown in Figure 4.31. Every Arduino program or sketch has three main parts:

1. Variable declaration section – variables that will be used in the program will be listed and declared in this section;
2. Setup () function – the setup () function will only run one time when the program start. The line of code or statements that set for actions to happen later on in the program are written in this function.; and
3. Loop () function – the loop () function will run over and over again until the program stopped or reprogrammed. The statements that perform the important action of the program are placed in this section.


localhost:4201/arduinocoding
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

CODING STRUCTURE - ARDUINO BOARD PROGRAMMING PROCESS

Basically, knowledge of the coding structure of the Arduino is important to know before start coding. Every Arduino program or sketch has three main parts:

1. Variable declaration section – variables that will be used in the program will be listed and declared in this section;

2. Setup () function – the setup () function will only run one time when the program start. The line of code or statements that set for actions to happen later on in the program are written in this function.; and

3. Loop () function – the loop () function will run over and over again until the program stopped or reprogrammed. The statements that perform the important action of the program are placed in this section.

Figure 4.31 : The approach screen on coding structure of Arduino board programming

Universiti Malaya

Meanwhile, there are 5 best practices as a guide to software developer while writing the programs for the Arduino board programming. The approach screen on the best practices is illustrated in Figure 4.32. The best practices are as listed:

- Best practices 1: C++ language knowledge – the software developers need to have the knowledge of the C++ language for writing the programs for the Arduino board programming. The C++ language is known as a common language in computing world. Rationale: By having the C++ language knowledge ease the coding process for Arduino board programming as the Arduino language is C++.
- Best practices 2: Basic syntax: The basic syntax that needs to be used in the programs are:

  a. Semicolon (;) - Every statement coded in the Arduino programming to end with a semicolon.

  b. Double backslash (//) - the double backslash is used for the single comment. Comments are used to explain the codes written. A good program should be commented.

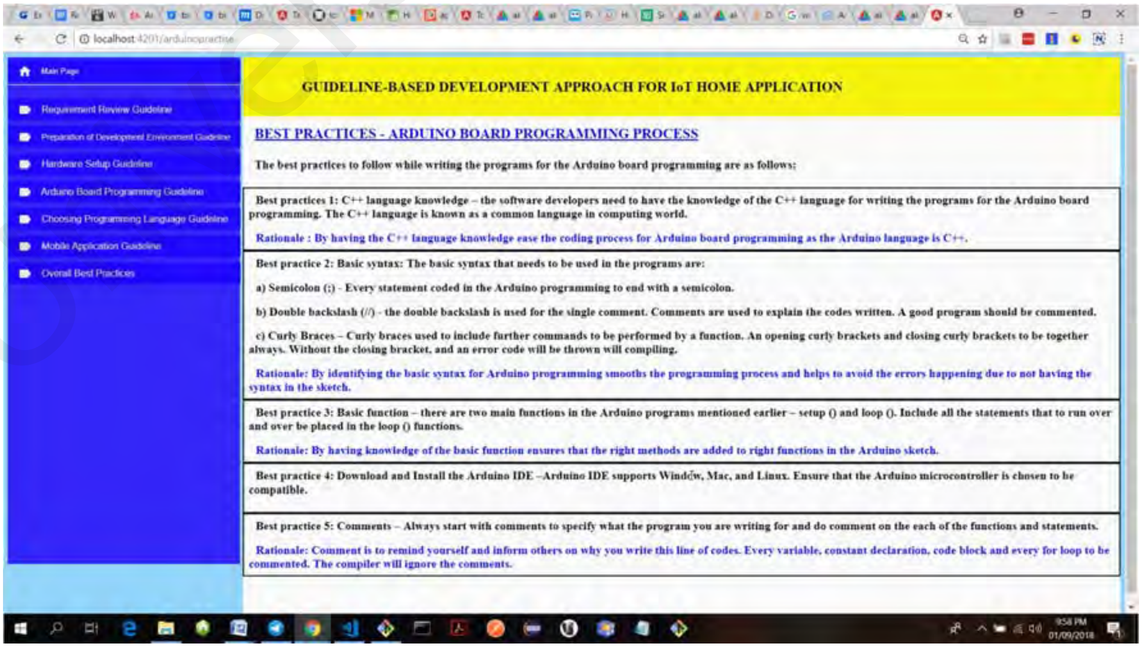
  c. Curly Braces ({}) – Curly braces used to include further commands to be performed by a function. An opening curly brackets and closing curly brackets to be together always. Without the closing bracket, and an error code will be thrown will compiling.

  Rationale: By identifying the basic syntax for Arduino programming smooths the programming process and helps to avoid the errors happening due to not having the syntax in the sketch.

Universiti Malaya

- Best practices 3: Basic function – there are two main functions in the Arduino programs mentioned earlier – setup () and loop (). Include all the statements that to run over and over be placed in the loop () functions. Rationale: By having knowledge of the basic function ensures that the right methods are added to right functions in the Arduino sketch.
- Best practices 4: Download and Install the Arduino IDE –Arduino IDE supports Window, Mac, and Linux. Ensure that the Arduino microcontroller is chosen to be compatible.
- Best practices 5: Comments – Always start with comments to specify what the program you are writing for and do comment on the each of the functions and statements. Rationale: Comment is to remind yourself and inform others on why you write this line of codes. Every variable, constant declaration, code block and every for loop to be commented. The compiler will ignore the comments.

Universiti Malaya


localhost:4201/arduinopractise
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices
9:58 PM
01/09/2018

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

BEST PRACTICES - ARDUINO BOARD PROGRAMMING PROCESS

The best practices to follow while writing the programs for the Arduino board programming are as follows:

Best practices 1: C++ language knowledge – the software developers need to have the knowledge of the C++ language for writing the programs for the Arduino board programming. The C++ language is known as a common language in computing world.

Rationale : By having the C++ language knowledge ease the coding process for Arduino board programming as the Arduino language is C++.

Best practice 2: Basic syntax: The basic syntax that needs to be used in the programs are:

a) Semicolon (;) - Every statement coded in the Arduino programming to end with a semicolon.

b) Double backslash (//) - the double backslash is used for the single comment. Comments are used to explain the codes written. A good program should be commented.

c) Curly Braces – Curly braces used to include further commands to be performed by a function. An opening curly brackets and closing curly brackets to be together always. Without the closing bracket, and an error code will be thrown will compiling.

Rationale: By identifying the basic syntax for Arduino programming smooths the programming process and helps to avoid the errors happening due to not having the syntax in the sketch.

Best practice 3: Basic function – there are two main functions in the Arduino programs mentioned earlier – setup () and loop (). Include all the statements that to run over and over be placed in the loop () functions.

Rationale: By having knowledge of the basic function ensures that the right methods are added to right functions in the Arduino sketch.

Best practice 4: Download and Install the Arduino IDE –Arduino IDE supports Window, Mac, and Linux. Ensure that the Arduino microcontroller is chosen to be compatible.

Best practice 5: Comments – Always start with comments to specify what the program you are writing for and do comment on the each of the functions and statements.

Rationale: Comment is to remind yourself and inform others on why you write this line of codes. Every variable, constant declaration, code block and every for loop to be commented. The compiler will ignore the comments.

Figure 4.32 : The approach screen on the best practices of Arduino board programming

4.6 Choosing programming language guidelines

The guideline choosing programming language will assist the software developers to decide on the suitable programming language to be adopted for the development, based on other factors such as the IDE chosen rather than depending on the programming language to choose the IDE. The guidelines for this section are in the form of best practices as follows:

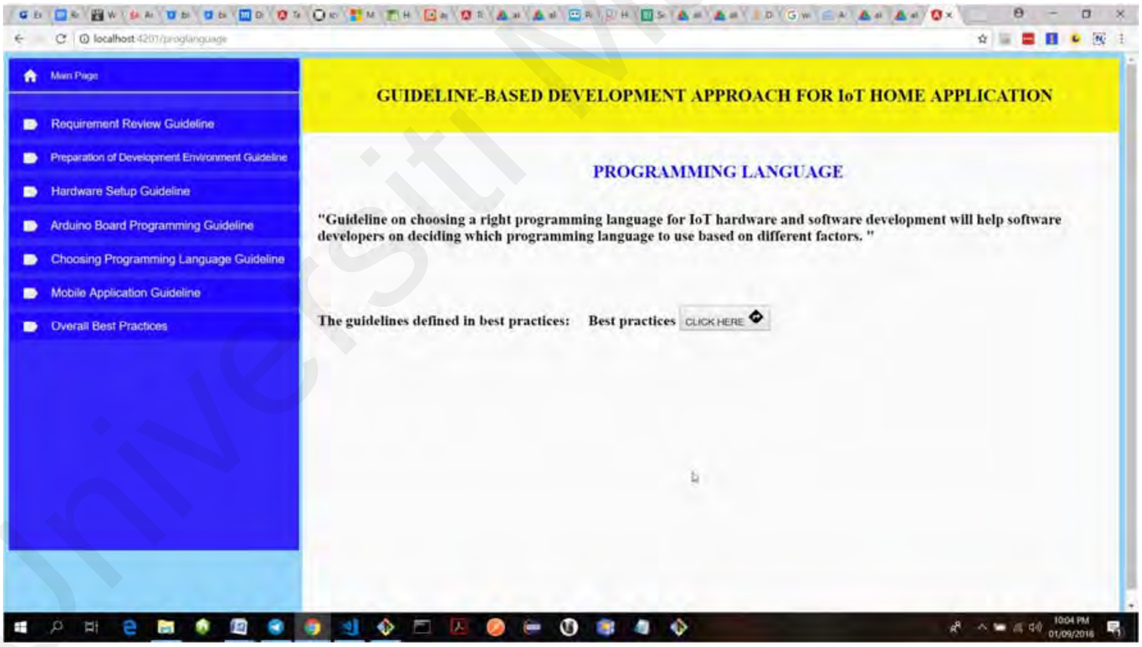
- Best practices 1: Based on the requirement of the project – Rationale: the project owner might have decided on the type of the programming language such as the requirement of the project is to use object-oriented. Therefore, the programming language to be used should be based on the requirement decided.
- Best practices 2: Operating system: The programming language can be chosen based on the mobile application operating system. Rationale: For example: if to develop an iOS mobile application, the programming language that is suitable for the iOS operating system is Objective-C and Swift language.
- Best practices 3: Open source – In the current world, there are many free open sources available for mobile application development and used by many software developers to reduce the development cost. Rationale: For example: for android mobile application development, Android studio is one of the free IDEs used regularly for the mobile application development and the programming language for the Android studio will be Java language.
- Best practices 4: The hardware and device to be used – The hardware and the smart devices for the IoT Home automation are also considered as a factor to decide on the programming language. Rationale: For example: if Arduino Uno is the microcontroller chosen for the development, then the developers need to download Arduino IDE and C++ language is the suitable language for the hardware development.

Universiti Malaya

- Best practices 5: Choose programming language based on popularity– Choosing a popular language for the mobile application. Rationale: By choosing a popular language for development, you might be able to stumble on the reference material, other helps and likely to find people to work together.

The main page of the choosing programming language approach screen is illustrated in Figure 4.33 and the approach screen on best practices for choosing programming language is shown in Figure 4.34.

Universiti Malaya


localhost:4201/proglanguage
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices
CLICK HERE

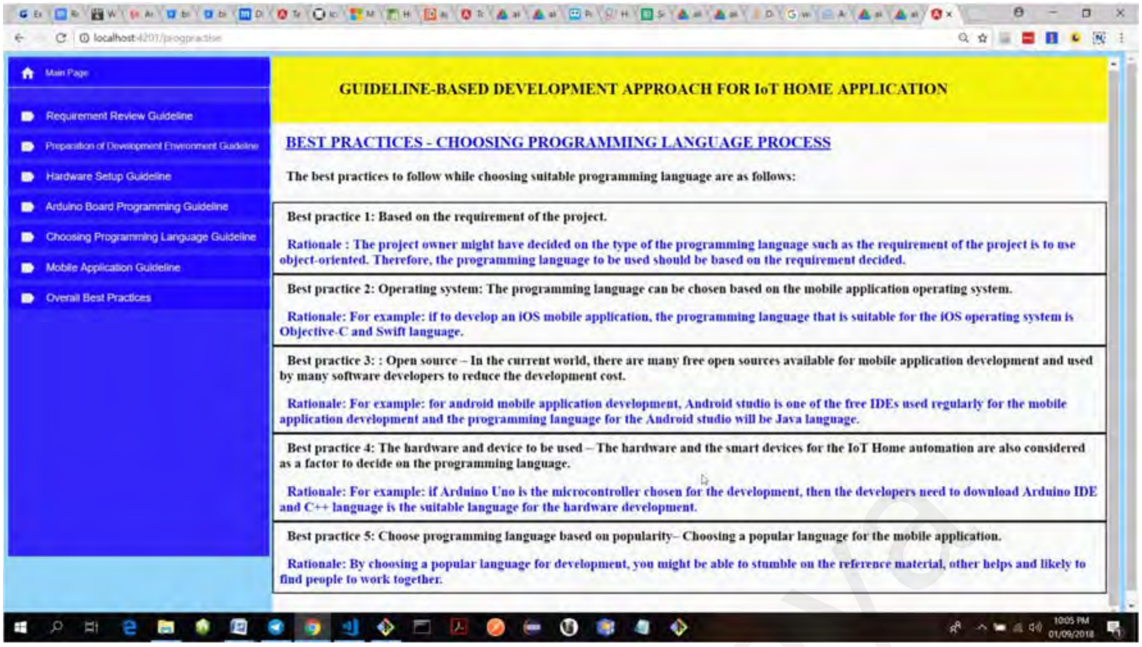
GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

PROGRAMMING LANGUAGE

"Guideline on choosing a right programming language for IoT hardware and software development will help software developers on deciding which programming language to use based on different factors. "

The guidelines defined in best practices: Best practices

Figure 4.33 : The approach screen of Choosing programming language main page



localhost:4201/progpractise
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

BEST PRACTICES - CHOOSING PROGRAMMING LANGUAGE PROCESS

The best practices to follow while choosing suitable programming language are as follows:

Best practice 1: Based on the requirement of the project.

Rationale : The project owner might have decided on the type of the programming language such as the requirement of the project is to use object-oriented. Therefore, the programming language to be used should be based on the requirement decided.

Best practice 2: Operating system: The programming language can be chosen based on the mobile application operating system.

Rationale: For example: if to develop an iOS mobile application, the programming language that is suitable for the iOS operating system is Objective-C and Swift language.

Best practice 3: : Open source – In the current world, there are many free open sources available for mobile application development and used by many software developers to reduce the development cost.

Rationale: For example: for android mobile application development, Android studio is one of the free IDEs used regularly for the mobile application development and the programming language for the Android studio will be Java language.

Best practice 4: The hardware and device to be used – The hardware and the smart devices for the IoT Home automation are also considered as a factor to decide on the programming language.

Rationale: For example: if Arduino Uno is the microcontroller chosen for the development, then the developers need to download Arduino IDE and C++ language is the suitable language for the hardware development.

Best practice 5: Choose programming language based on popularity– Choosing a popular language for the mobile application.

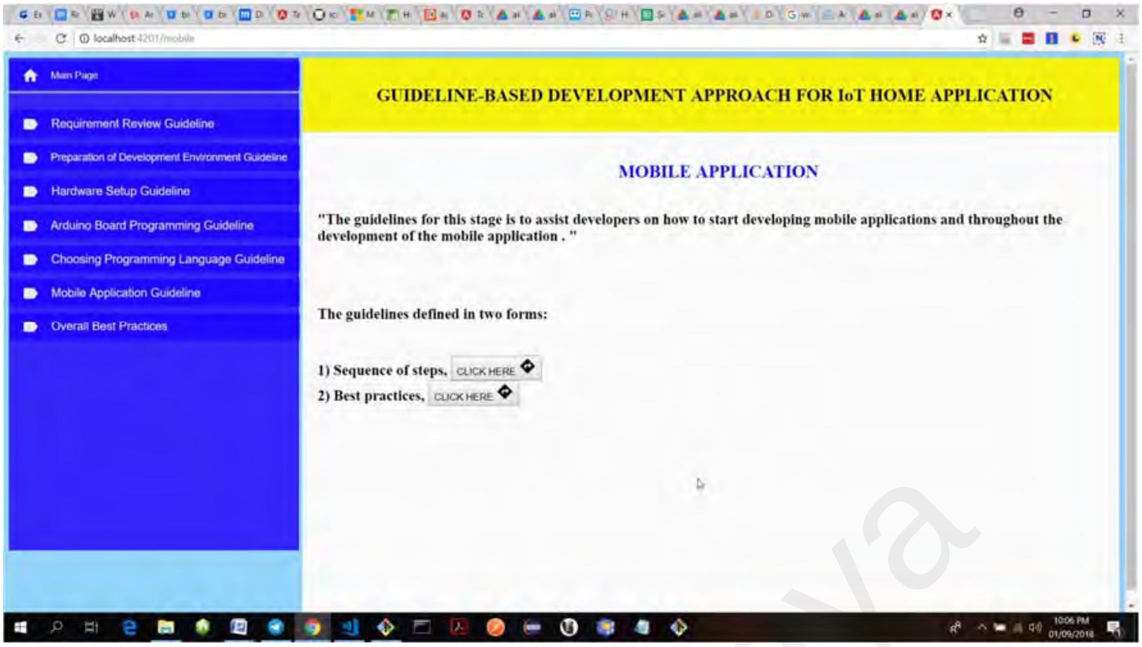
Rationale: By choosing a popular language for development, you might be able to stumble on the reference material, other helps and likely to find people to work together.

Figure 4.34 : The approach screen on the best practices of choosing programming language process

4.7 Mobile application development guidelines

Mobile application development will be the last stage of the development process for the IoT Home application. The mobile application development guidelines will assist the software developers on how to begin the mobile application development and the best practices to apply while doing development the same. The guidelines for this mobile application development will be in the form of best practices and sequence of steps. The approach screen on the guideline of mobile application development is shown in Figure 4.35.

Universiti Malaya



Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

MOBILE APPLICATION

"The guidelines for this stage is to assist developers on how to start developing mobile applications and throughout the development of the mobile application . "

The guidelines defined in two forms:

1) Sequence of steps, CLICK HERE

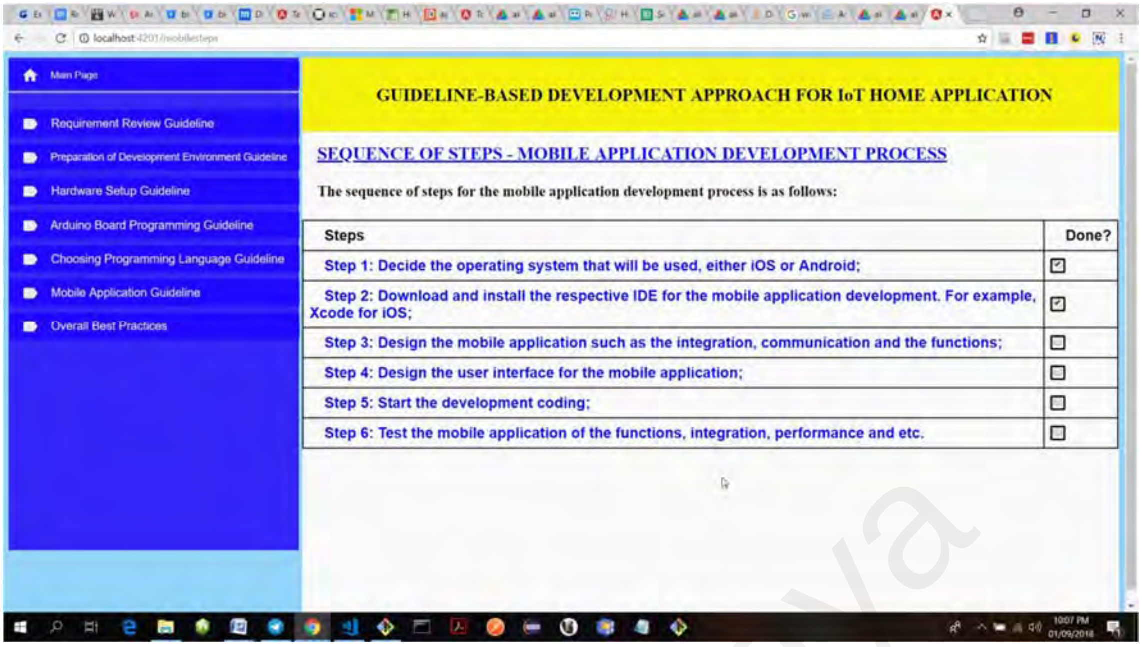
2) Best practices, CLICK HERE

Figure 4.35 : The guideines of mobile application main page of approach screen

Universiti Malaya

The steps proposed for this process are as listed:

1. Decide the operating system that will be used, either iOS or Android;
2. Download and install the respective IDE for the mobile application development. For example, Xcode for iOS.
3. Design the mobile application such as the integration, communication and the functions;
4. Design the user interface for the mobile application;
5. Start the development coding;
6. Test the mobile application of the functions, integration, performance and etc.



Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

SEQUENCE OF STEPS - MOBILE APPLICATION DEVELOPMENT PROCESS

The sequence of steps for the mobile application development process is as follows:

Steps	Done?
Step 1: Decide the operating system that will be used, either iOS or Android;	☑
Step 2: Download and install the respective IDE for the mobile application development. For example, Xcode for iOS;	☑
Step 3: Design the mobile application such as the integration, communication and the functions;	□
Step 4: Design the user interface for the mobile application;	□
Step 5: Start the development coding;	□
Step 6: Test the mobile application of the functions, integration, performance and etc.	□

Figure 4.36 : The approach screen on the sequence of steps in mobile application development process

Universiti Malaya

Figure 4.36 shows the approach screen on the sequence of steps and Figure 4.37 shows the approach screen on the best practices of mobile application development process.

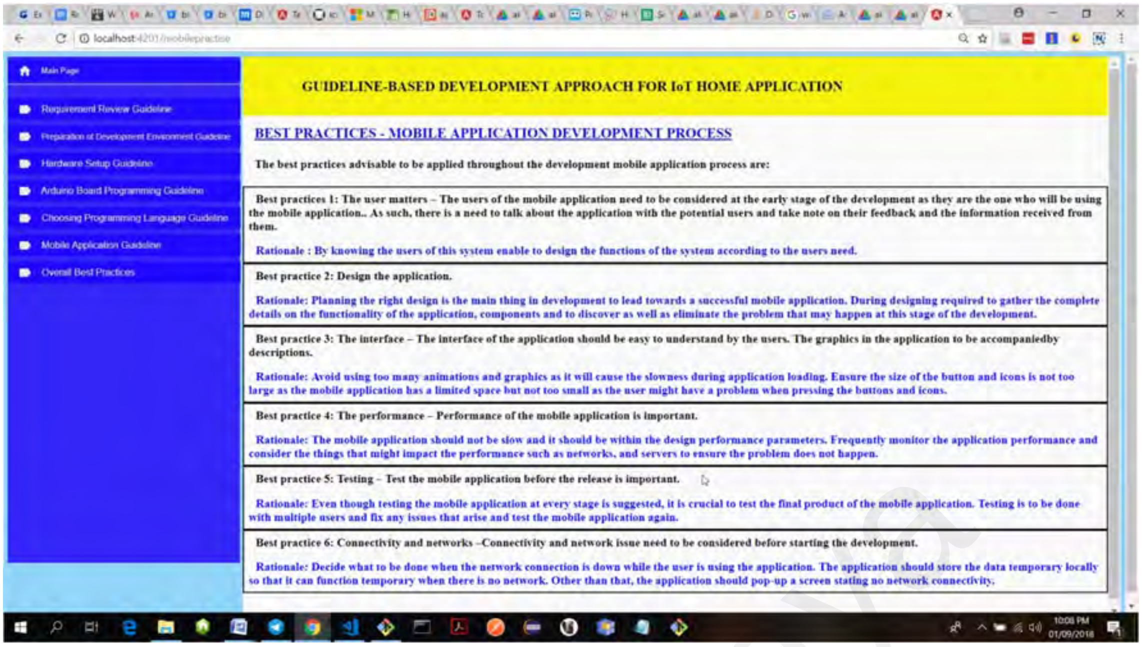
Best practices advisable to be applied throughout the development mobile application process are:

- Best practices 1: The user matters – The users of the mobile application need to be considered at the early stage of the development as they are the one who will be using the mobile application.. As such, there is a need to talk about the application with the potential users and take note on their feedback and the information received from them. Rationale: By knowing the users of this system enable to design the functions of the system according to the users need.
- Best practices 2: Design the application: Rationale: Planning the right design is the main thing in development to lead towards a successful mobile application.

During designing required to gather the complete details on the functionality of the application, components and to discover as well as eliminate the problem that may happen at this stage of the development.

- Best practices 3: The interface – The interface of the application should be easy to understand by the users. The graphics in the application to be accompaniedby descriptions. Rationale: Avoid using too many animations and graphics as it will cause the slowness during application loading. Ensure the size of the button and icons is not too large as the mobile application has a limited space but not too small as the user might have a problem when pressing the buttons and icons.
- Best practices 4: The performance – Performance of the mobile application is important. Rationale: The mobile application should not be slow and it should be within the design performance parameters. Frequently monitor the application performance and consider the things that might impact the performance such as networks, and servers to ensure the problem does not happen.
- Best practices 5: Testing – Test the mobile application before the release is important. Rationale: Even though testing the mobile application at every stage is suggested, it is crucial to test the final product of the mobile application. Testing is to be done with multiple users and fix any issues that arise and test the mobile application again.
- Best practices 6: Connectivity and networks –Connectivity and network issue need to be considered before starting the development. Rationale: Decide what to be done when the network connection is down while the user is using the application. The application should store the data temporary locally so that it can function temporary when there is no network. Other than that, the application should pop-up a screen stating no network connectivity.

Universiti Malaya


localhost:4201/mobilepractise
Main Page
Requirement Review Guideline
Preparation of Development Environment Guideline
Hardware Setup Guideline
Arduino Board Programming Guideline
Choosing Programming Language Guideline
Mobile Application Guideline
Overall Best Practices
10:06 PM
01/09/2018

GUIDELINE-BASED DEVELOPMENT APPROACH FOR IoT HOME APPLICATION

BEST PRACTICES - MOBILE APPLICATION DEVELOPMENT PROCESS

The best practices advisable to be applied throughout the development mobile application process are:

Best practices 1: The user matters – The users of the mobile application need to be considered at the early stage of the development as they are the one who will be using the mobile application.. As such, there is a need to talk about the application with the potential users and take note on their feedback and the information received from them.

Rationale : By knowing the users of this system enable to design the functions of the system according to the users need.

Best practice 2: Design the application.

Rationale: Planning the right design is the main thing in development to lead towards a successful mobile application. During designing required to gather the complete details on the functionality of the application, components and to discover as well as eliminate the problem that may happen at this stage of the development.

Best practice 3: The interface – The interface of the application should be easy to understand by the users. The graphics in the application to be accompaniedby descriptions.

Rationale: Avoid using too many animations and graphics as it will cause the slowness during application loading. Ensure the size of the button and icons is not too large as the mobile application has a limited space but not too small as the user might have a problem when pressing the buttons and icons.

Best practice 4: The performance – Performance of the mobile application is important.

Rationale: The mobile application should not be slow and it should be within the design performance parameters. Frequently monitor the application performance and consider the things that might impact the performance such as networks, and servers to ensure the problem does not happen.

Best practice 5: Testing – Test the mobile application before the release is important.

Rationale: Even though testing the mobile application at every stage is suggested, it is crucial to test the final product of the mobile application. Testing is to be done with multiple users and fix any issues that arise and test the mobile application again.

Best practice 6: Connectivity and networks –Connectivity and network issue need to be considered before starting the development.

Rationale: Decide what to be done when the network connection is down while the user is using the application. The application should store the data temporary locally so that it can function temporary when there is no network. Other than that, the application should pop-up a screen stating no network connectivity.

Figure 4.37 : The approach screen on best practices of mobile application development process

Universiti Malaya

4.8 Chapter Summary

This chapter proposed and discussed guideline-based development approach for the six stages of IoT Home application development. The guidelines in the form of best practice, pattern, checklist, and sequence of steps were proposed for six different stages or process involved throughout the IoT Home application. The approach proposed is to be applied by the software developers and as guidance during the development process of IoT Home application. The guidelines are proposed for the requirement review process, development environment preparation process, hardware setup process, Arduino board programming process, choosing the suitable language process, and mobile application development process.

The next chapter will discuss the design and development of IoT Home application by applying the proposed approach in this chapter.

CHAPTER 5: SYSTEM DESIGN AND DEVELOPMENT

5.1 Introduction

Guideline-based development approach is presented in the previous chapter with the objective to help and assist in the IoT home application development process. The proposed approach for the IoT home application is to be applied throughout the development process.

This chapter presents the details of the IoT home application system developed by applying the guidelines in the approach proposed in Chapter 4. A software development lifecycle has the following phases that must be followed in order to complete the software development, which are: 1) requirement gathering and analysis, 2) System analysis, 3) Coding, 4) Testing, and 5) Implementation. Meanwhile this IoT home application development will involve six phases, as follows: 1) requirement analysis or review phase, 2) hardware and software development environment preparation phase, 3) hardware setting up phase, 4) Arduino board coding phase, 5) mobile application coding phase, and 6) Testing phase. The proposed guideline-based development approach will be applied to the first five phases of this IoT home application development process.

Each detail of the development process for the IoT home application by applying the proposed guideline-based development approach is discussed in the following sections.

Universiti Malaya

5.2 Requirements analysis or review phase

Requirements analysis or review phase is the first phase in an IoT Home application development process after communication with the project owner on the requirements of the project. At this phase, the development team will look at the requirement details, analyze and review the requirements. Furthermore, the required information is gathered and unclear requirements were confirmed with the project owner to enable the development team to begin the development environment set up and coding process.

5.2.1 Requirements analysis and review process for IoT home application system

First of all, the proposed sequence of steps was adapted for this requirement analysis and review process: -

- Step 1: Went through the requirements document for this IoT home application system;
- Step 2: Reviewed the requirements document by going through each of the requirements;
- Step 3: Conducted the requirements review process by cross-checking the details with the checklist that was proposed in the earlier chapter. Figure 5.1 shows the checklist details for the IoT home application system;
- Step 4: The missing required information is identified and updated in the requirements document;
- Step 5: The ambiguous information is identified and confirmed the status of the information;
- Step 6: Finalized the requirements document details.

Universiti Malaya

No.	Details	Checked	Comments
1.	Business information is adequate.	√	
2.	Requirements details are complete.	√	
3.	Requirements are compatible.	√	
4.	Performance and environment information are adequate.	√	
5.	The user requirements are adequately stated.	√	
6.	The response time of the system function is specified.	√	
7.	All the functional requirements stated by the client.	√	
8.	All the non-functional requirements stated by the client.	√	
9.	The hardware resources are decided and described.	√	
10.	The external software for the hardware and the mobile application are decided and described.	√	
11.	The interface of the system is discussed and confirmed.	√	
12.	Each function and requirement is testable.	√	
13.	The doable future changes on the system are specified.	√	

Figure 5.1 : The requirements review checklist for IoT home application system

Universiti Malaya

5.2.2 The best practices applied to this IoT home application system

The best practices proposed were applied in the requirements review process for IoT home application system as follows: -

- Best practice 1: Conducted the requirements review process.
- Best practice 2: The software design specification document was prepared for this IoT home application system.
- Best practice 3: The system's functionality was breakdown to sub-functionality.
- Best practice 4: The requirements document for this system was regularly updated when a new requirement was recognized and changes made.

5.3 Hardware and software development environment preparation phase

Hardware and software development environment preparation phase is subsequent to requirements analysis and review phase. In this phase, the required hardware and software components for the IoT home application system development were prepared. The development team arranged the hardware components, coding for the hardware components and wrote programs for the mobile application. The guidelines in approach proposed for this phase were applied and the details were discussed further in the next section.

Universiti Malaya

5.3.1 Development environment preparation for IoT home application system

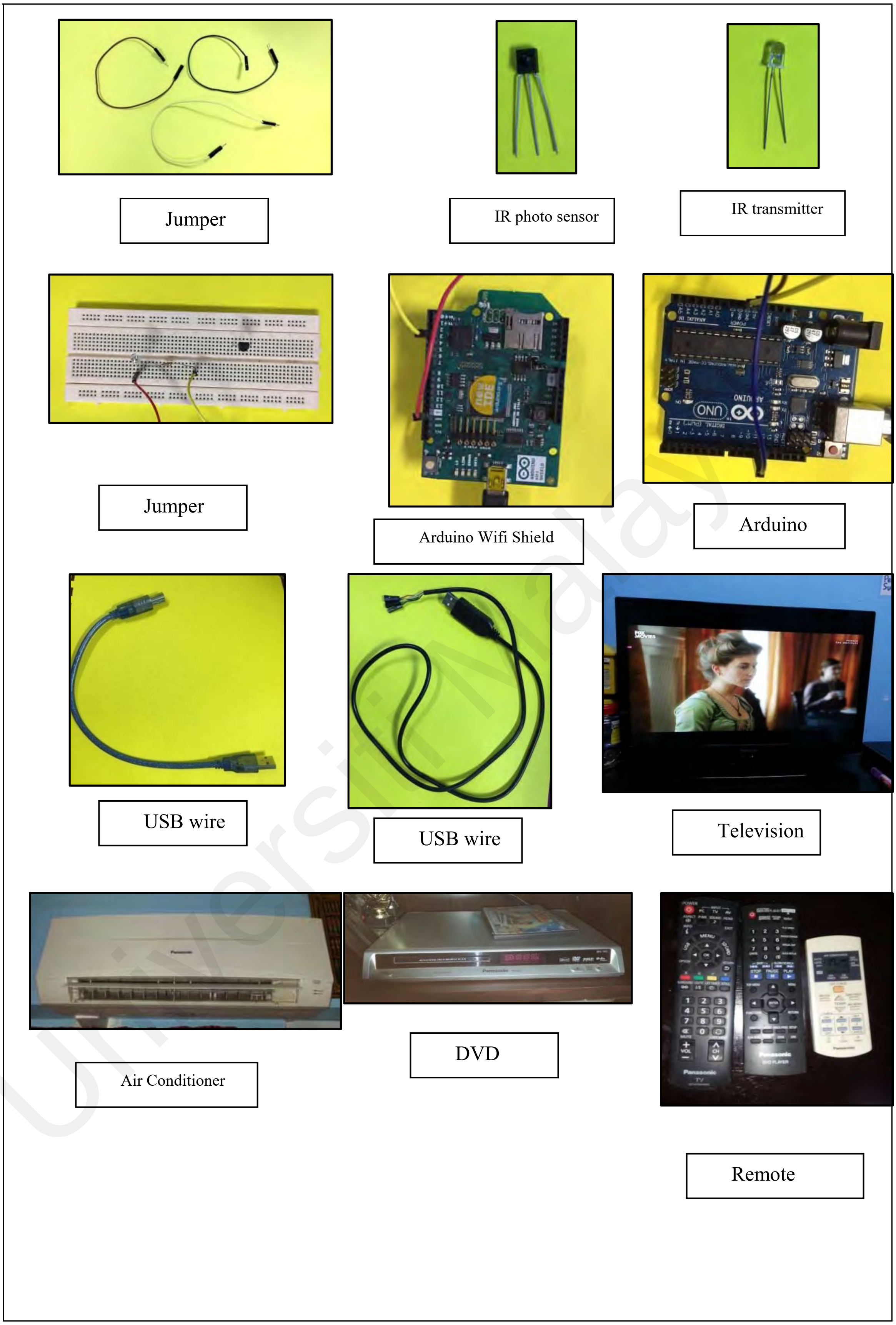
The sequence of steps proposed as a guideline was applied to assist in this phase is as follows:

1. Identified and prepared the hardware components and smart devices for the IoT home application development which are: 1) Arduino Uno, 2) Arduino Wifi Shield, 3) IR photo sensor, 4) IR transmitter, 5) Jumper wire, 6) Bread board, 7) USB wires, 8) Laptop, and 9) Smart devices (TV, DVD player, and Air Conditioner) and remote controls. The hardware components and smart devices are shown in Figure 5.2.
2. Identified and prepared the required software for the hardware components mentioned in step 1 which are: 1) Arduino IDE, and 2) Library required for IR communication – IR Remote Library;

3. Identified and prepared the required hardware components for Android and IOS mobile applications, which are: 1) Laptop for Android and IOS mobile application development, 2) Android mobile phone, and 3) IOS mobile phone;
4. Identified and prepared the required software for Android and IOS mobile application system, which are: 1) Android Studio IDE, 2) Xcode, and 3) Java Runtime Environment (JRE); and
5. Prepared the network connection, Wi-Fi and 3G.

While attending to the above steps, a checklist is used as guidance to ensure hardware or software required for the IoT home application development are not left out. Figure 5.3 shows the development environment preparation checklist for the IoT home application development.

Universiti Malaya



Universiti Malaya

Figure 5.2: The hardware components and smart devices used for IoT home application development

No.	Details	Checked	Comments
1	The hardware required parts	√	
a	Arduino Microcontroller	√	
b	Arduino for wifi communication	√	
c	Sensors	√	
d	Bread Board	√	
e	IR transmitter and IR receiver	√	
f	Jumper wire	√	
g	Smart devices (e.g. Tv, dvd player, air conditioner)	√	
2	The required software for the hardware	√	
a	Arduino IDE	√	
3	The required hardware for mobile application	√	
a	Laptop or desktop for android application	√	
b	Apple laptop or iMac for IOS application	√	
c	iPhone and Android phone	√	
4	The required software for the mobile application.	√	
a	Android IDE	√	
b	macOS IDE - Xcode	√	
c	Java Runtime Environment (JRE)	√	
5	The connectivity part (Network)	√	

Figure 5.3 : The checklist for IoT home application system development environment preparation

Universiti Malaya

5.3.2 The best practices applied to the development environment preparation phase

The best practices proposed were adapted in this phase of the IoT home application development process as follows:

- Best practice 1: Manage the data received and sent. Identified the data that will be received from the mobile application and sent to the IR transmitter to be fowarded to the smart devices. As such, the IR sensors and the Arduino microcontroller were chosen to be used.

- Best practice 2: Hardware platform. The number of customers is less than 100 in a house and the smart devices are the home appliances consisting of TV, DVD player and Air Conditioner. As such the Arduino microcontroller is chosen to be used for this system.
- Best practice 3: Smart things or smart devices. Identified the smart devices involved, which are: TV, DVD player and Air Conditioner.
- Best practice 4: The software. Have chosen the suitable programming language based on the guideline in approach proposed for choosing the suitable programming language in the previous chapter. They are Arduino Board Programming – C++ language, Android mobile application – Java language, and IOS mobile application – swift language.
- Best practice 5: Connectivity issue. Tested and ensure the WiFi connection and 3G connection are available for development.
- Best practices 6: IoT development kits. Decided the suitable software and hardware development kits that are compatible with Windows, Linux, and Mac. The software and hardware development kits are Arduino IDE and Android Studio.

Universiti Malaya

5.4 Hardware setting up phase

Hardware setting up phase is where the microcontroller and other hardware components were wired or connected together to function as a receiver, and demodulation of IR signal as well as transmission of IR signal to smart devices. This phase is where the real challenge for any software developer who does not have electrical and electronic experience that the guideline on the approach proposed for this stage has helped to smooth the process of setting up the hardware components. The guideline adapted for this phase is discussed in detail in the following sections.5.3.1 Development environment preparation for IoT home application system.

- **Hardware setup for IoT home application system**

The sequence of steps recommended for hardware setup phase was applied and the details are as follows:

1. Receiving and decoding IR signals from the remote control of Panasonic TV, Panasonic DVD player, and Panasonic Air Conditioner.

a. Wired IR photo sensor to Arduino UNO by using Jumper wire and breadboard – Pin 1 of IR photo sensor to pin 11 at Arduino Uno, Pin 2 of IR photo sensor to Ground at Arduino Uno, and Pin 3 of IR photo sensor to GV at Arduino Uno.

b. Downloaded IR library for IR remotes.

c. The IrrecvDump file was modified, compiled and uploaded in order to begin encoding of the signals from the remote control.

d. Started encoding the remote control's signal by pointing the remote control to the IR photo sensor and the values were displayed at the open serial monitor. Figure 5.4 shows the encoding process using remote control.

The hardware setup for IR signal decoding is shown in Figure 5.5

Universiti Malaya

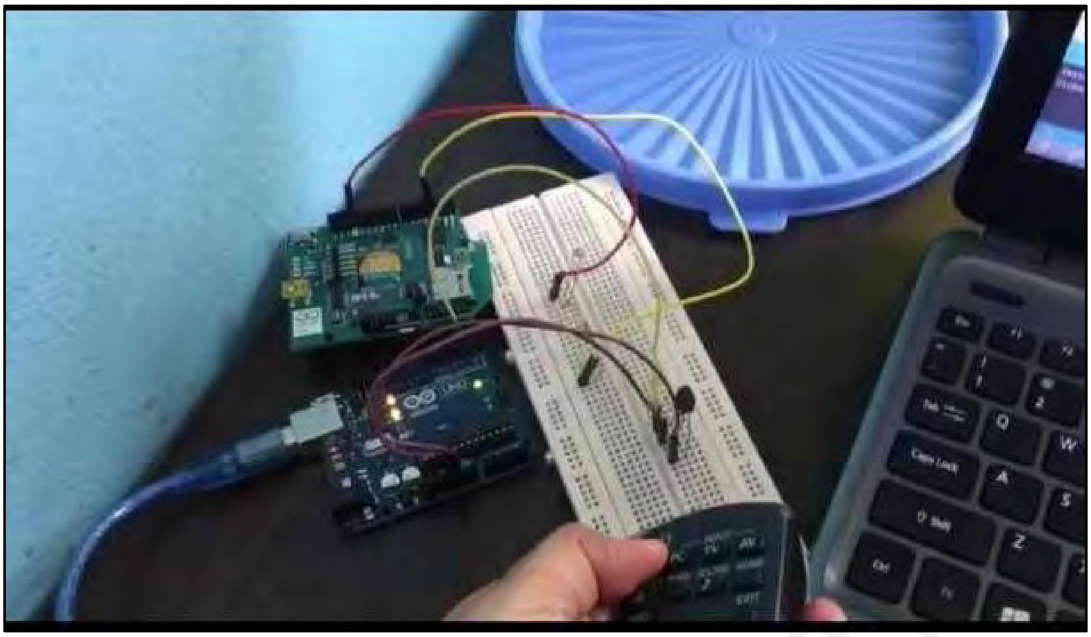


Figure 5.4 : The encoding process using remote control

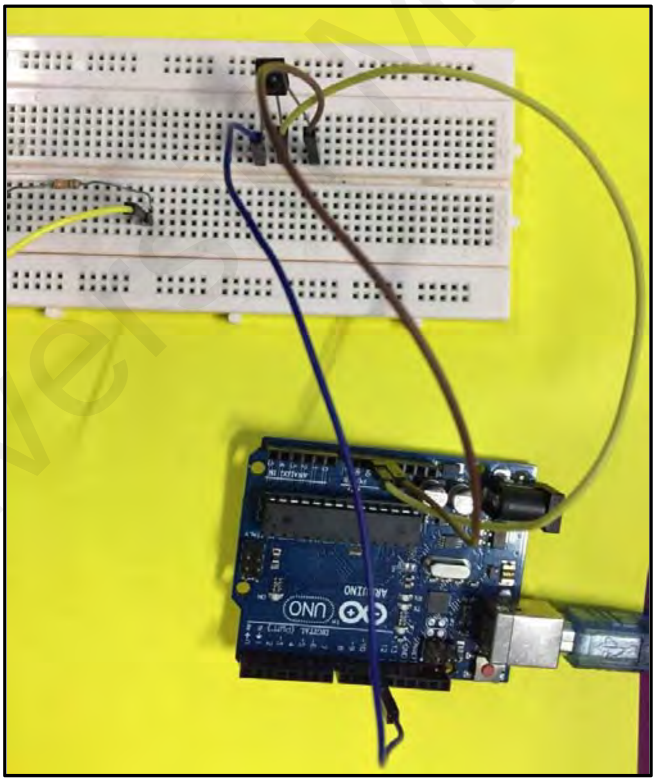


Figure 5.5 : The hardware setup for decoding IR Signal

University of Malaya

2. Sending IR signals from IR transmitter to TV, DVD player and Air conditioner.

a. The Arduino Wifi Shield was mounted up on top of Arduino Uno

b. Wired IR transmitter which is the IR LED to Arduino Wifi Shield by using Jumper wire and breadboard – Pin 1 of IR LED to Ground at Arduino Wifi Shield, and Pin 2 of IR LED to Pin 3 at Arduino Wifi Shield.

c. The coded sketch sends the signals from IR LED to home appliances was compiled and uploaded. (The details of the sketch that was coded and uploaded will be discussed in the next section).

The hardware setup for transmitting IR signal to the smart device is shown in Figure 5.6.

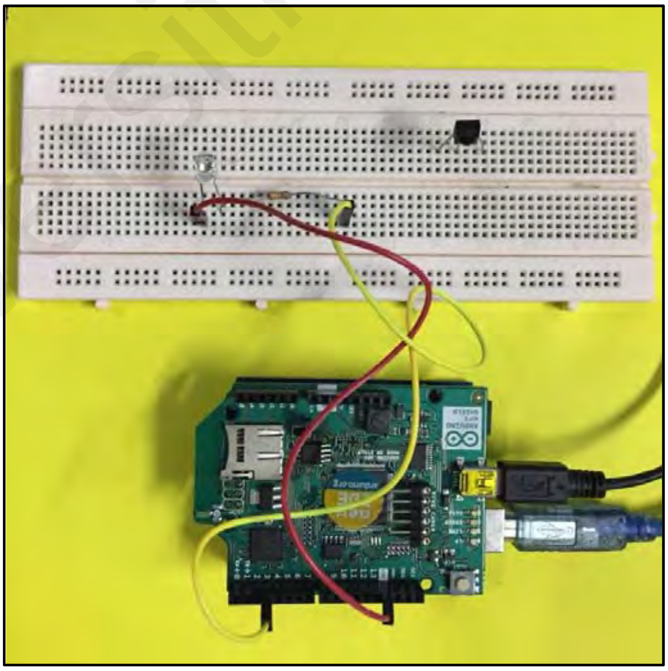


Figure 5.6 : The hardware setup for transmitting IR Signal

Universiti Malaya

Figure 5.7 shows the transmitting IR signal from the mobile app to the smart device.

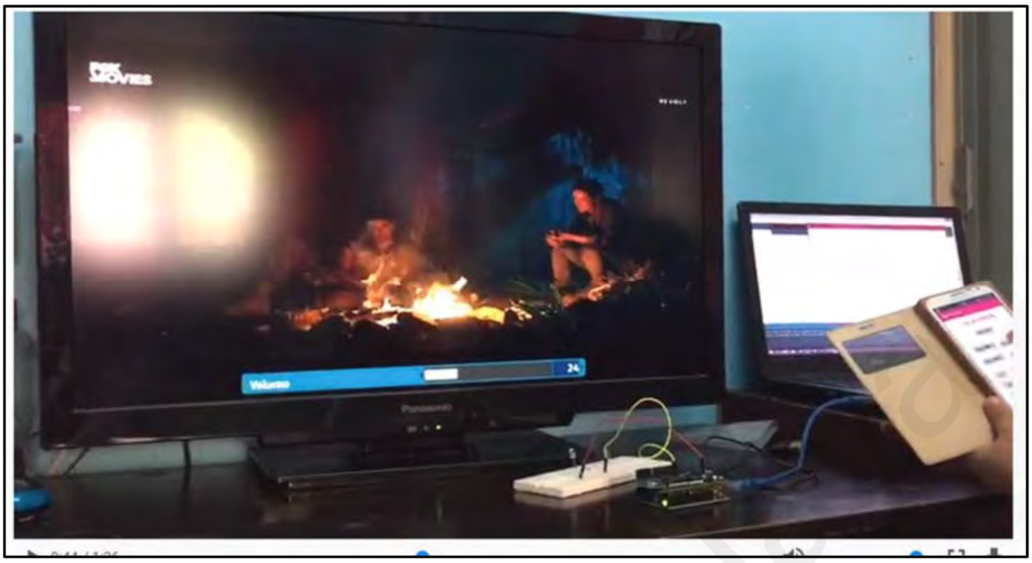


Figure 5.7 : The IR signal transmitting to smart device from smart phone

5.5 Arduino board coding phase

The coding for decoding the IR Signal and sending IR of signal is executed in this phase. The basic sequence of steps was followed before beginning the coding process.

The steps that were followed are:

1. Downloaded and installed Arduino IDE from https://www.arduino.cc/en/Main/Software.
2. Selected the Arduino Uno for board selection in the tool menu of Arduino IDE.
3. Downloaded the IR remote library and Wifi library for the Arduino WiFi 101 Shield.
4. For the decoding IR Signal – IrrecvDemo.ino from the IR remote library was compiled an uploaded to Arduino Uno for the decoding process. Figure 5.8

Universiti Malaya

shows the code or sketch used for the decoding process which is downloaded from https://github.com/z3t0/Arduino-IRremote and Figure 5.9 shows the Arduino IDE with the sketch for decoding. Figure 5.10 shows the serial monitor that displays the values from the decoding process.

5. For the sending of IR Signal – codes are written in a new sketch. Figure 5.11 shows the code or sketch used for transmitting the IR signal to smart devices and Figure 5.12 shows the Arduino IDE with the sketch for transmitting signal.

Universiti Malaya

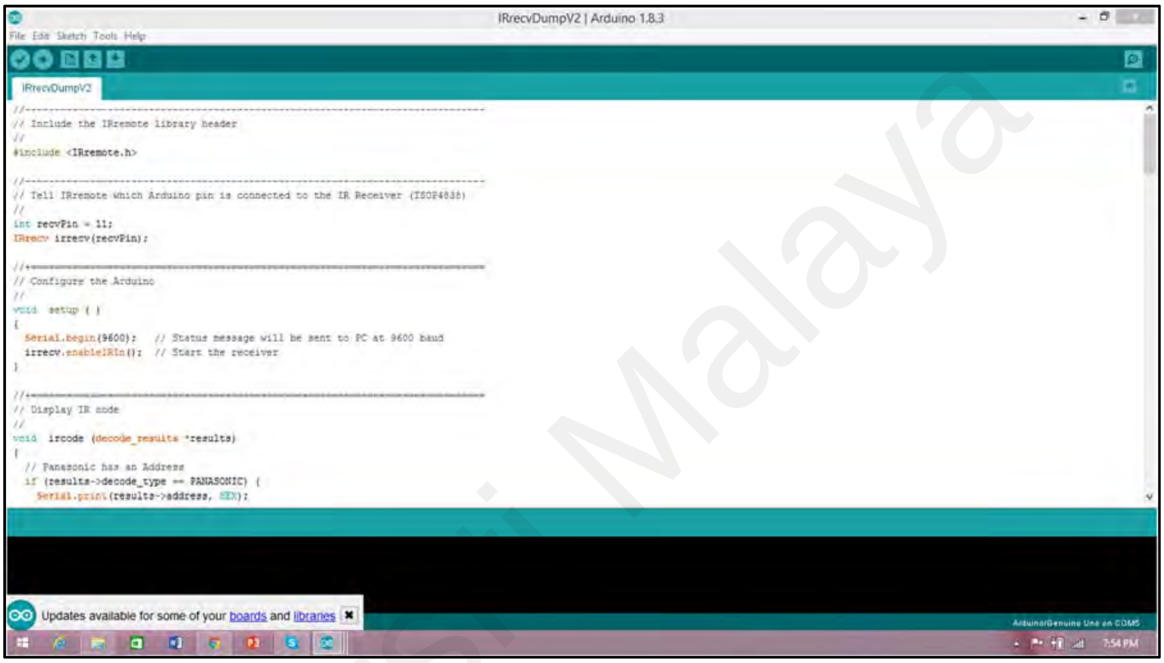
```
//------------------------------------------------------------------------------
// Include the IRremote library header
//
#include <IRremote.h>
//------------------------------------------------------------------------------
// Tell IRremote which Arduino pin is connected to the IR Receiver (TSOP4838)
//
intrecvPin = 11;
IRrecvirrecv(recvPin);
//+=============================================================================
// Configure the Arduino
//
void  setup ( )
{
Serial.begin(9600);   // Status message will be sent to PC at 9600 baud
irrecv.enableIRIn();  // Start the receiver
}
//+=============================================================================
// Display IR code
//
void  ircode (decode_results *results)
{
  // Panasonic has an Address
  if (results->decode_type == PANASONIC) {
Serial.print(results->address, HEX);
Serial.print(":");
  }
  // Print Code
Serial.print(results->value, HEX);
}
//+=============================================================================
// Display encoding type
//
void  encoding (decode_results *results)
{
  switch (results->decode_type) {
    default:
    case UNKNOWN:      Serial.print("UNKNOWN");       break ;
    case NEC:          Serial.print("NEC");           break ;
    case SONY:         Serial.print("SONY");          break ;
    case RC5:          Serial.print("RC5");           break ;
    case RC6:          Serial.print("RC6");           break ;
    case DISH:         Serial.print("DISH");          break ;
    case SHARP:        Serial.print("SHARP");         break ;
    case JVC:          Serial.print("JVC");           break ;
    case SANYO:        Serial.print("SANYO");         break ;
    case MITSUBISHI:   Serial.print("MITSUBISHI");    break ;
    case SAMSUNG:      Serial.print("SAMSUNG");       break ;
    case LG:           Serial.print("LG");            break ;
    case WHYNTER:      Serial.print("WHYNTER");       break ;
    case AIWA_RC_T501: Serial.print("AIWA_RC_T501");  break ;
    case PANASONIC:    Serial.print("PANASONIC");     break ;
    case DENON:        Serial.print("Denon");         break ;
  }
}
//+=============================================================================
// Dump out the decode_results structure.
```

```
//
void  dumpInfo (decode_results *results)
{
  // Check if the buffer overflowed
  if (results->overflow) {
Serial.println("IR code too long. Edit IRremoteInt.h and increase RAWLEN");
    return;
  }
  // Show Encoding standard
Serial.print("Encoding  : ");
  encoding(results);
Serial.println("");
  // Show Code & length
Serial.print("Code      : ");
ircode(results);
Serial.print(" (");
Serial.print(results->bits, DEC);
Serial.println(" bits)");
}
//+=============================================================================
// Dump out the decode_results structure.
//
void  dumpRaw (decode_results *results)
{
  // Print Raw data
Serial.print("Timing[");
Serial.print(results->rawlen-1, DEC);
Serial.println("]: ");
  for (inti = 1;  i< results->rawlen;  i++) {
    unsigned long  x = results->rawbuf[i] * USECPERTICK;
    if (!(i& 1)) {  // even
Serial.print("-");
      if (x < 1000)  Serial.print(" ") ;
      if (x < 100)   Serial.print(" ") ;
Serial.print(x, DEC);
    } else {  // odd
Serial.print("     ");
Serial.print("+");
      if (x < 1000)  Serial.print(" ") ;
      if (x < 100)   Serial.print(" ") ;
Serial.print(x, DEC);
      if (i< results->rawlen-1) Serial.print(", "); //',' not needed for last one
    }
    if (!(i % 8))  Serial.println("");
  }
Serial.println("");                    // Newline
}
//+=============================================================================
// Dump out the decode_results structure.
//
void  dumpCode (decode_results *results)
{
  // Start declaration
Serial.print("unsigned int  ");          // variable type
Serial.print("rawData[");                // array name
Serial.print(results->rawlen - 1, DEC);  // array size
Serial.print("] = {");                   // Start declaration
  // Dump data
  for (inti = 1;  i< results->rawlen;  i++) {
Serial.print(results->rawbuf[i] * USECPERTICK, DEC);
    if ( i< results->rawlen-1 ) Serial.print(","); // ',' not needed on last one
    if (!(i& 1))  Serial.print(" ");
  }
  // End declaration
Serial.print("};");  //
  // Comment
Serial.print("  // ");
  encoding(results);
Serial.print(" ");
ircode(results);
  // Newline
Serial.println("");
  // Now dump "known" codes
  if (results->decode_type != UNKNOWN) {
    // Some protocols have an address
    if (results->decode_type == PANASONIC) {
Serial.print("unsigned intaddr = 0x");
Serial.print(results->address, HEX);
Serial.println(";");
    }
    // All protocols have data
Serial.print("unsigned int  data = 0x");
Serial.print(results->value, HEX);
Serial.println(";");
  }
}
//+=============================================================================
// The repeating section of the code
//
```

Universiti Malaya

```
void  loop ( )
{
decode_results  results;        // Somewhere to store the results
  if (irrecv.decode(&results)) {  // Grab an IR code
dumpInfo(&results);           // Output the results
dumpRaw(&results);            // Output the results in RAW format
dumpCode(&results);           // Output the results as source code
Serial.println("");         // Blank line between entries
irrecv.resume();            // Prepare for the next value
  }
}
```

Figure 5.8: The sketch or code used for the decoding process

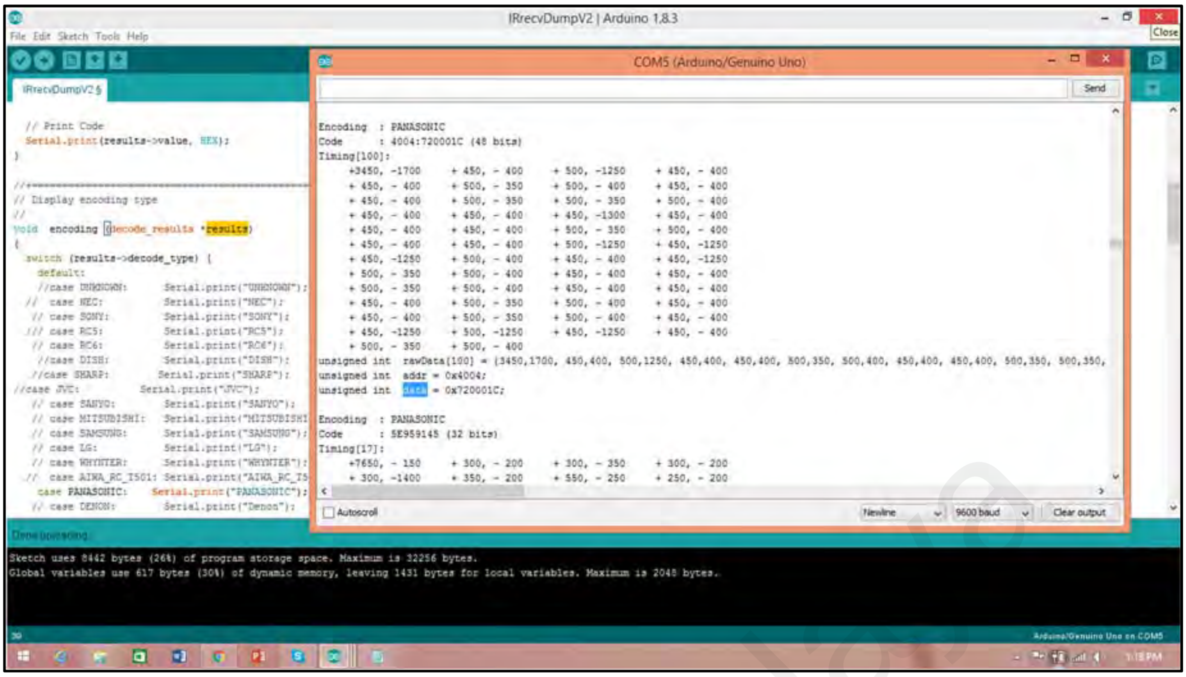

IRrecvDumpV2 | Arduino 1.8.3
File Edit Sketch Tools Help
IRrecvDumpV2
Arduino/Genuine Uno on COM5
7:54 PM

```
//------------------------------------------------------------------------------
// Include the IRremote library header
//
#include <IRremote.h>

//------------------------------------------------------------------------------
// Tell IRremote which Arduino pin is connected to the IR Receiver (TSOP4838)
//
int recvPin = 11;
IRrecv irrecv(recvPin);

//+=============================================================================
// Configure the Arduino
//
void  setup ( )
{
  Serial.begin(9600);   // Status message will be sent to PC at 9600 baud
  irrecv.enableIRIn();  // Start the receiver
}

//+=============================================================================
// Display IR code
//
void  ircode (decode_results *results)
{
  // Panasonic has an Address
  if (results->decode_type == PANASONIC) {
    Serial.print(results->address, HEX);
```

Updates available for some of your boards and libraries

Figure 5.9 : Arduino IDE with the sketch for the decoding process

Universiti Malaya


IRrecvDumpV2 | Arduino 1.8.3
File Edit Sketch Tools Help
IRrecvDumpV2 §
COM5 (Arduino/Genuino Uno)
Send
Autoscroll
Newline
9600 baud
Clear output
Done uploading.
Close

```
  // Print Code
  Serial.print(results->value, HEX);
}

//+=============================================================
// Display encoding type
//
void  encoding (decode_results *results)
{
  switch (results->decode_type) {
    default:
    //case UNKNOWN:      Serial.print("UNKNOWN");
  //  case NEC:          Serial.print("NEC");
   // case SONY:         Serial.print("SONY");
  /// case RC5:          Serial.print("RC5");
   // case RC6:          Serial.print("RC6");
    //case DISH:         Serial.print("DISH");
   //case SHARP:        Serial.print("SHARP");
//case JVC:          Serial.print("JVC");
   // case SANYO:        Serial.print("SANYO");
   // case MITSUBISHI:   Serial.print("MITSUBISHI
   // case SAMSUNG:      Serial.print("SAMSUNG");
   // case LG:           Serial.print("LG");
   // case WHYNTER:      Serial.print("WHYNTER");
 //  case AIWA_RC_T501: Serial.print("AIWA_RC_T5
    case PANASONIC:    Serial.print("PANASONIC");
   // case DENON:        Serial.print("Denon");
```

```
Encoding  : PANASONIC
Code      : 4004:720001C (48 bits)
Timing[100]:
     +3450, -1700     + 450, - 400     + 500, -1250     + 450, - 400
     + 450, - 400     + 500, - 350     + 500, - 400     + 450, - 400
     + 450, - 400     + 500, - 350     + 500, - 350     + 500, - 400
     + 450, - 400     + 450, - 400     + 450, -1300     + 450, - 400
     + 450, - 400     + 450, - 400     + 500, - 350     + 500, - 400
     + 450, - 400     + 450, - 400     + 500, -1250     + 450, -1250
     + 450, -1250     + 500, - 400     + 450, - 400     + 450, -1250
     + 500, - 350     + 500, - 400     + 450, - 400     + 450, - 400
     + 500, - 350     + 500, - 400     + 450, - 400     + 450, - 400
     + 450, - 400     + 500, - 350     + 500, - 400     + 450, - 400
     + 450, - 400     + 500, - 350     + 500, - 400     + 450, - 400
     + 450, -1250     + 500, -1250     + 450, -1250     + 450, - 400
     + 500, - 350     + 500, - 400
unsigned int  rawData[100] = {3450,1700, 450,400, 500,1250, 450,400, 450,400, 500,350, 500,400, 450,400, 450,400, 500,350, 500,350,
unsigned int  addr = 0x4004;
unsigned int  data = 0x720001C;

Encoding  : PANASONIC
Code      : 5E959145 (32 bits)
Timing[17]:
     +7650, - 150     + 300, - 200     + 300, - 350     + 300, - 200
     + 300, -1400     + 350, - 200     + 550, - 250     + 250, - 200
```

```
Sketch uses 8442 bytes (26%) of program storage space. Maximum is 32256 bytes.
Global variables use 617 bytes (30%) of dynamic memory, leaving 1431 bytes for local variables. Maximum is 2048 bytes.
```

Figure 5.10 : Serial monitor displays the values from the decoding process

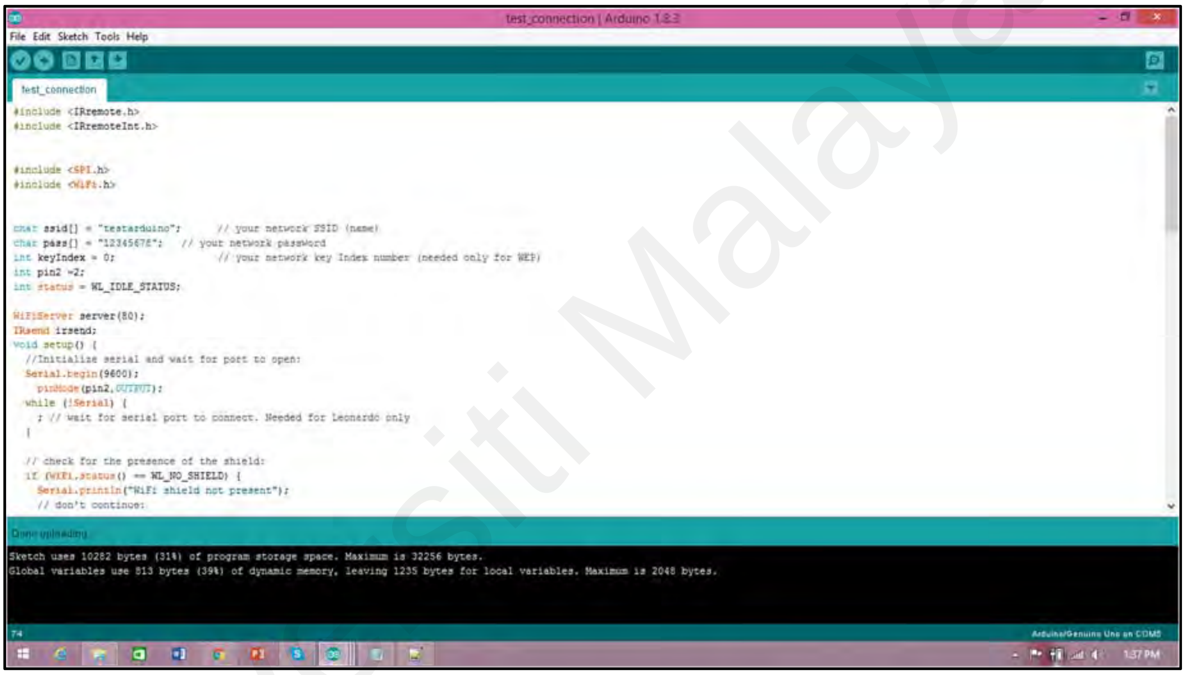
Universiti Malaya

```
#include <IRremote.h>
#include <IRremoteInt.h>
#include <SPI.h>
#include <WiFi.h>
char ssid[] = "testarduino";     // your network SSID (name)
char pass[] = "12345678";   // your network password
intkeyIndex = 0;                 // your network key Index number (needed only for WEP)
int pin2 =2;
int status = WL_IDLE_STATUS;
WiFiServer server(80);
IRsendirsend;
void setup() {
  //Initialize serial and wait for port to open:
Serial.begin(9600);
  // pinMode(pin2,OUTPUT);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for Leonardo only
  }
// check for the presence of the shield:
  if (WiFi.status() == WL_NO_SHIELD) {
Serial.println("WiFi shield not present");
    // don't continue:
    while(true);
  }
  // attempt to connect to Wifi network:
  while ( status != WL_CONNECTED) {
Serial.print("Attempting to connect to SSID: ");
Serial.println(ssid);
    // Connect to WPA/WPA2 network. Change this line if using open or WEP network:
    status = WiFi.begin(ssid, pass);
    // wait 10 seconds for connection:
    delay(10000);
  }
server.begin();
  // you're connected now, so print out the status:
printWifiStatus();
}
void loop() {
  // listen for incoming clients
WiFiClient client = server.available();
  if (client) {
Serial.println("new client");
    // an http request ends with a blank line
booleancurrentLineIsBlank = true;
    String buffered = "";
    while (client.connected()) {
      if (client.available()) {
        char c = client.read();
Serial.write(c);
        buffered +=c;
   if(buffered.endsWith("O")){
      //digitalWrite(pin2,HIGH);
irsend.sendPanasonic(0x4004,0x100BCBD);
      //delay(100);
    }
   if (buffered.endsWith("X")) {
irsend.sendPanasonic(0x4004,0x1004C4D);
    }
   if (buffered.endsWith("P")) {
irsend.sendPanasonic(0x4004,0x1000405);
     }
   if (buffered.endsWith("M")) {
irsend.sendPanasonic(0x4004,0x1008485);
     }
   if (buffered.endsWith("1")) {
irsend.sendPanasonic(0x4004,0x1000809);
     }
   if (buffered.endsWith("2")) {
irsend.sendPanasonic(0x4004,0x1008889);
     }
   if (buffered.endsWith("3")) {
irsend.sendPanasonic(0x4004,0x1004849);
     }
   if (buffered.endsWith("4")) {
irsend.sendPanasonic(0x4004,0x100C8C9);
     }
   if (buffered.endsWith("5")) {
irsend.sendPanasonic(0x4004,0x1002829);
     }
   if (buffered.endsWith("6")) {
irsend.sendPanasonic(0x4004,0x100A8A9);
     }
   if (buffered.endsWith("7")) {
irsend.sendPanasonic(0x4004,0x1006869);
     }
   if (buffered.endsWith("8")) {
irsend.sendPanasonic(0x4004,0x100E8E9);
     }
   if (buffered.endsWith("9")) {
irsend.sendPanasonic(0x4004,0x1001819);
     }
   if (buffered.endsWith("0")) {
irsend.sendPanasonic(0x4004,0x1009899);
     }
   if (buffered.endsWith("CP")) {
irsend.sendPanasonic(0x4004,0x1002C2D);
     }
   if (buffered.endsWith("CM")) {
irsend.sendPanasonic(0x4004,0x100ACAD);
     }
   if (buffered.endsWith("ON")) {
irsend.sendPanasonic(0x4004,0x720001C);
     }
   if (buffered.endsWith("DOP")) {  //this is for DVD PLAYER
irsend.sendPanasonic(0x4004,0xD00808D);
     }
   if (buffered.endsWith("DLT")) {  // this is for DVD PLAYER
irsend.sendPanasonic(0x4004,0xD00202D);
     }
   if (buffered.endsWith("DRT")) {  // this is for DVD PLAYER
irsend.sendPanasonic(0x4004,0xD00A0AD);
     }
   if (buffered.endsWith("DPY")) {  // this is for DVD PLAYER
irsend.sendPanasonic(0x4004,0xD00505D);
     }
   if (buffered.endsWith("DPU")) { // this is for DVD PLAYER
irsend.sendPanasonic(0x4004,0xD00606D);
     }
   if (buffered.endsWith("DSP")) { // this is for DVD PLAYER
irsend.sendPanasonic(0x4004,0xD00000D);
     }
   if (buffered.endsWith("DMN")) { // this is for DVD PLAYER
irsend.sendPanasonic(0x4004,0xD00010C);
     }
   if (buffered.endsWith("DER")) { // this is for DVD PLAYER
irsend.sendPanasonic(0x4004,0xD00414C);
     }
   if (buffered.endsWith("DRN")) { // this is for DVD PLAYER
irsend.sendPanasonic(0x4004,0xD00818C);
     }
   if (buffered.endsWith("DON")) { // this is for DVD PLAYER
irsend.sendPanasonic(0x4004,0xD00BCB1);
     }
       }
    }
    // give the web browser time to receive the data
    delay(1);
```

Universiti Malaya

```
    // close the connection:
client.stop();
Serial.println("client disonnected");
  }
}
void printWifiStatus() {
  // print the SSID of the network you're attached to:
Serial.print("SSID: ");
Serial.println(WiFi.SSID());
  // print your WiFi shield's IP address:
IPAddressip = WiFi.localIP();
Serial.print("IP Address: ");
Serial.println(ip);
  // print the received signal strength:
  long rssi = WiFi.RSSI();
Serial.print("signal strength (RSSI):");
Serial.print(rssi);
Serial.println(" dBm");
}
```

Figure 5.11: The sketch or code used for transmitting IR signal



```
#include <IRremote.h>
#include <IRremoteInt.h>


#include <SPI.h>
#include <WiFi.h>


char ssid[] = "testarduino";      // your network SSID (name)
char pass[] = "12345678";   // your network password
int keyIndex = 0;                 // your network key Index number (needed only for WEP)
int pin2 =2;
int status = WL_IDLE_STATUS;

WiFiServer server(80);
IRsend irsend;
void setup() {
  //Initialize serial and wait for port to open:
  Serial.begin(9600);
    pinMode(pin2,OUTPUT);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for Leonardo only
  }

  // check for the presence of the shield:
  if (WiFi.status() == WL_NO_SHIELD) {
    Serial.println("WiFi shield not present");
    // don't continue:
```

Done uploading.

```
Sketch uses 10282 bytes (31%) of program storage space. Maximum is 32256 bytes.
Global variables use 813 bytes (39%) of dynamic memory, leaving 1235 bytes for local variables. Maximum is 2048 bytes.
```

Figure 5.12 : Arduino IDE with the sketch for transmitting IR signal process

The proposed best practices were applied and have helped to ease the coding process for Arduino board. The best practices are:

- Best practices 1: C++ language knowledge – Went through the available tutorial to get knowledge of the C++ programming language.
- Best practices 2: Basic syntax: The basic syntax listed are used in the Arduino programming and the information on each basic syntax have helped on identifying how and when to use them.

Universiti Malaya

- Best practices 3: Basic function – Identified and written the suitable methods that run inside the basic function which is to send the IR signal in loop () functions.
- Best practices 4: Download and Install the Arduino IDE – Downloaded Arduino IDE and chose Arduino Uno as it supports Window, Mac, and Linux.
- Best practices 5: Comments – Included comments as a reminder or information on the methods and variables.

Universiti Malaya

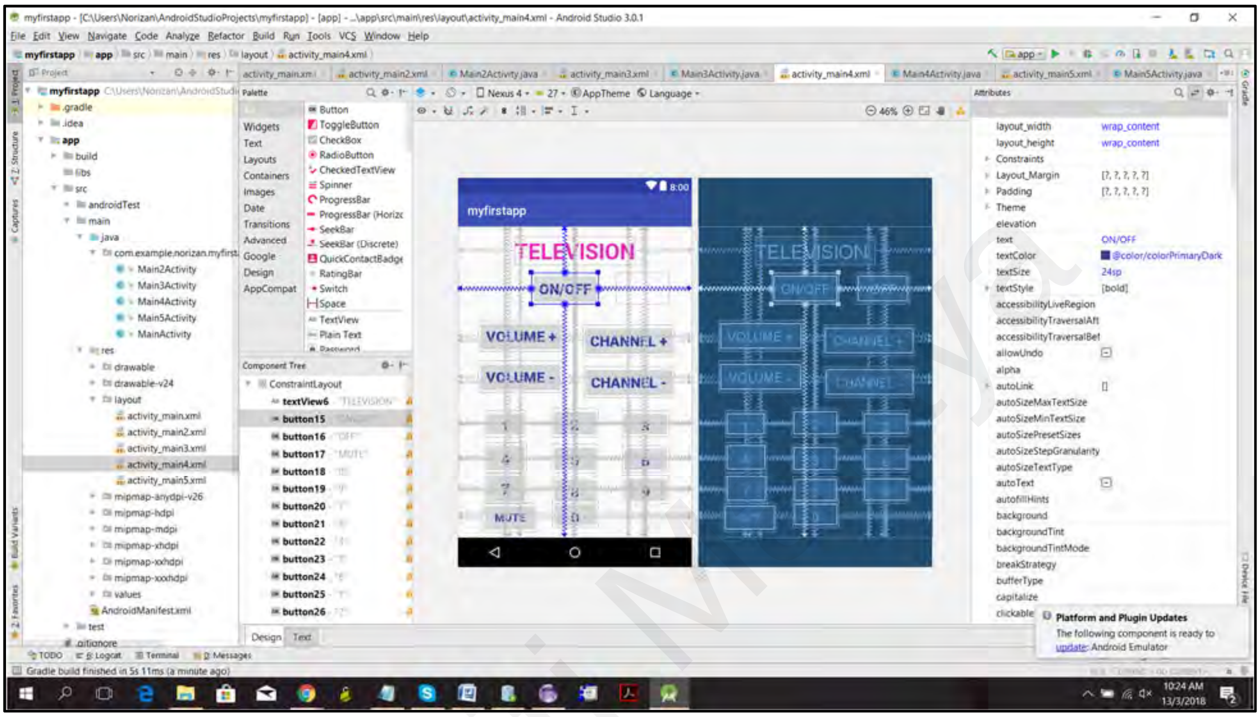
5.6 Mobile application coding phase

A sequence of steps assists in developing Android and iOS mobile applications. The steps that were followed are divided into Android mobile application and IOS mobile application development.

The steps that were applied to the Android mobile application are:

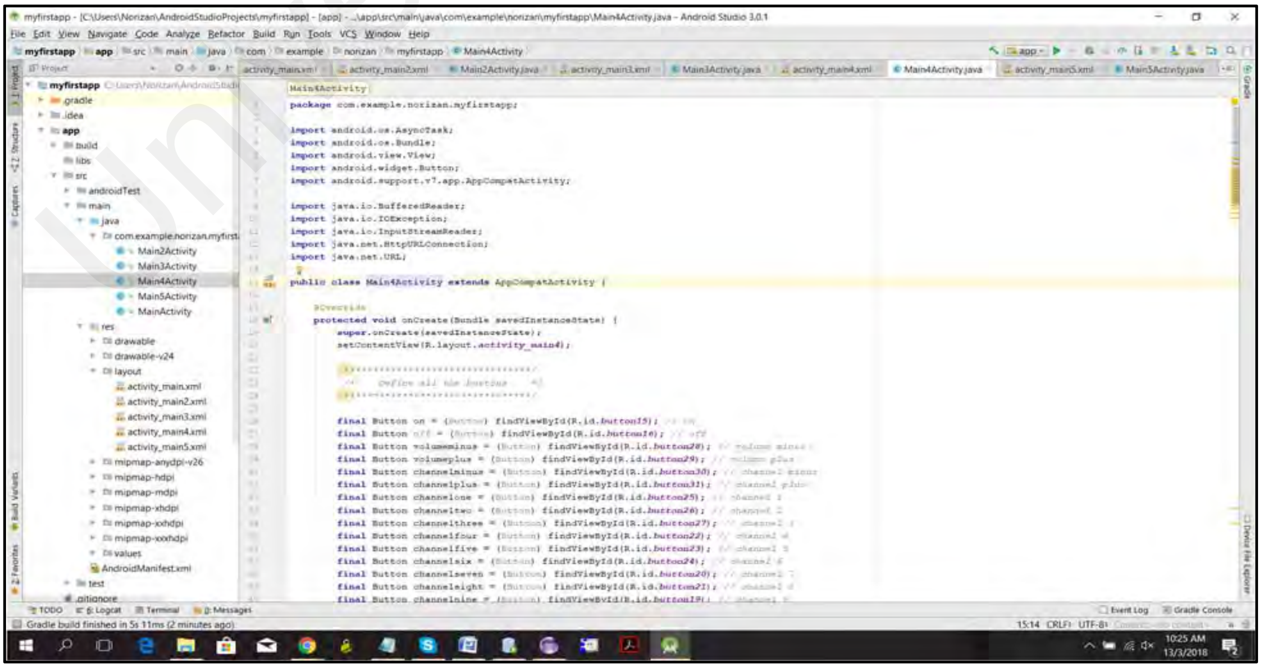
1. Download and install Android Studio IDE.
2. Chose the Java programming language based on the guideline in approach proposed for choosing the suitable programming language;
3. Designed Android mobile application integration, and the functions;
4. Designed the user interface;
5. Coded the methods to assign the values and send the values to Arduino by Get Method;
6. Tested the Android mobile application functions, integrations, usability, and performance.

Figure 5.13, 5.14 and 5.15 show the Android Studio IDE with development screens. Meanwhile, Figure 5.16 shows the user interface of Android mobile application after the completion of development process for Android mobile application.



```
layout_width        wrap_content
layout_height       wrap_content
Layout_Margin       [?, ?, ?, ?]
Padding             [?, ?, ?, ?]
text                ON/OFF
textColor           @color/colorPrimaryDark
textSize            24sp
textStyle           [bold]
```

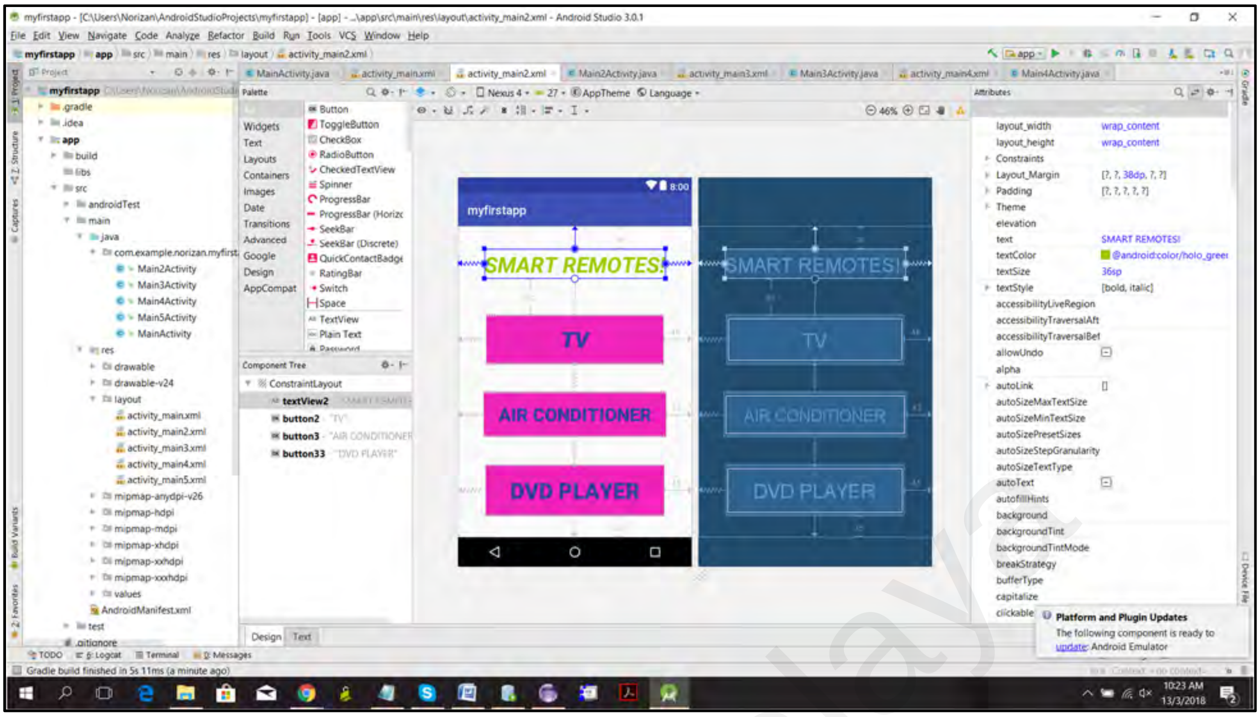
Figure 5.13 : Android Studio IDE with development screen for television remote control



```java
package com.example.norizan.myfirstapp;

import android.os.AsyncTask;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.support.v7.app.AppCompatActivity;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;

public class Main4Activity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main4);

        /*******************************/
        /*    Define all the buttons    */
        /*******************************/

        final Button on = (Button) findViewById(R.id.button15); // on
        final Button off = (Button) findViewById(R.id.button16); // off
        final Button volumeminus = (Button) findViewById(R.id.button28); // volume minus
        final Button volumeplus = (Button) findViewById(R.id.button29); // volume plus
        final Button channelminus = (Button) findViewById(R.id.button30); // channel minus
        final Button channelplus = (Button) findViewById(R.id.button31); // channel plus
        final Button channelone = (Button) findViewById(R.id.button25); // channel 1
        final Button channeltwo = (Button) findViewById(R.id.button26); // channel 2
        final Button channelthree = (Button) findViewById(R.id.button27); // channel 3
        final Button channelfour = (Button) findViewById(R.id.button22); // channel 4
        final Button channelfive = (Button) findViewById(R.id.button23); // channel 5
        final Button channelsix = (Button) findViewById(R.id.button24); // channel 6
        final Button channelseven = (Button) findViewById(R.id.button20); // channel 7
        final Button channeleight = (Button) findViewById(R.id.button21); // channel 8
        final Button channelnine = (Button) findViewById(R.id.button19); // channel 9
```

Figure 5.14 : Android Studio IDE with java file for television remote control

Universiti Malaya



myfirstapp - [C:\Users\Norizan\AndroidStudioProjects\myfirstapp] - [app] - ...\app\src\main\res\layout\activity_main2.xml - Android Studio 3.0.1

Component Tree

- ConstraintLayout
  - textView2 - "SMART REMOTES!"
  - button2 - "TV"
  - button3 - "AIR CONDITIONER"
  - button33 - "DVD PLAYER"

Attributes

layout_width	wrap_content
layout_height	wrap_content
Constraints	
Layout_Margin	[?, ?, 38dp, ?, ?]
Padding	[?, ?, ?, ?, ?]
Theme	
elevation	
text	SMART REMOTES!
textColor	🟩 @android:color/holo_greer
textSize	36sp
textStyle	[bold, italic]

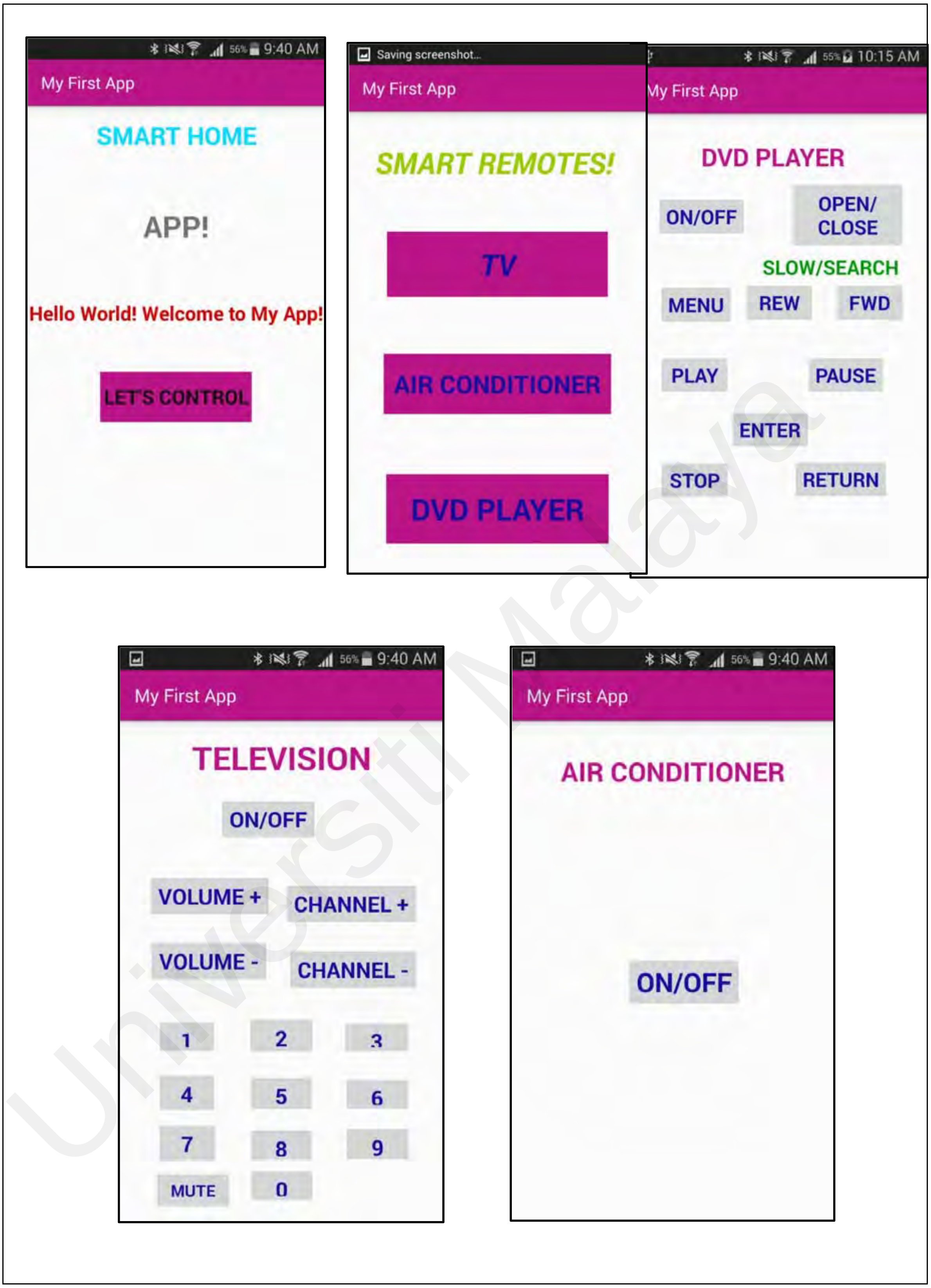
Platform and Plugin Updates

The following component is ready to update: Android Emulator

Gradle build finished in 5s 11ms (a minute ago)

Figure 5.15 : Android Studio IDE with development screen for menu smart remotes

Universiti Malaya



Universiti Malaya

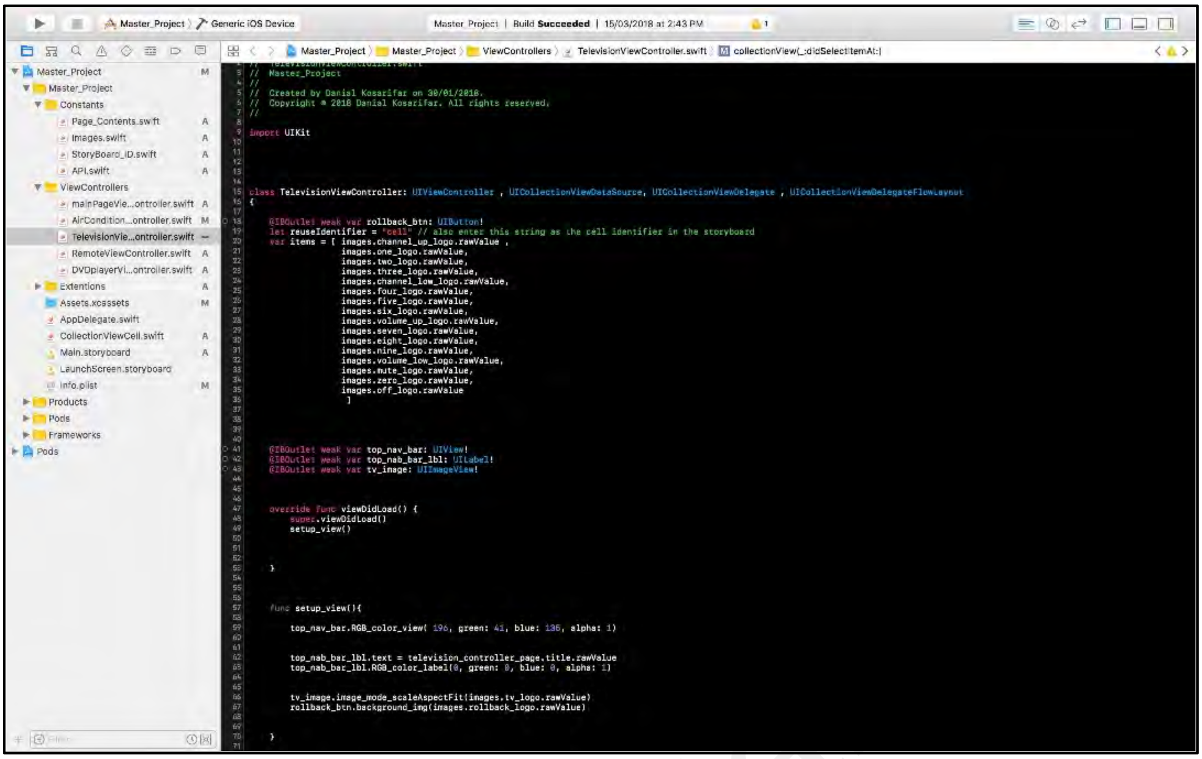
Figure 5.16: User Interface of Android mobile application for IoT home application

The steps that were applied to the IOS mobile application are:

1. Download and install Xcode;
2. Chose swift programming language based on the guideline in approach proposed for choosing the suitable programming language;
3. Designed IOS mobile application integration, and the functions;
4. Designed the user interface;
5. Coded the methods to assign the values and send the values to Arduino by Get Method;
6. Tested the IOS mobile application functions, integrations, usability, and performance.

Figure 5.17, 5.18 and 5.19 shows the Xcode IDE with development screens. Meanwhile, Figure 5.20 shows the user interface of the iOS mobile application after the completion of iOS mobile application development process.

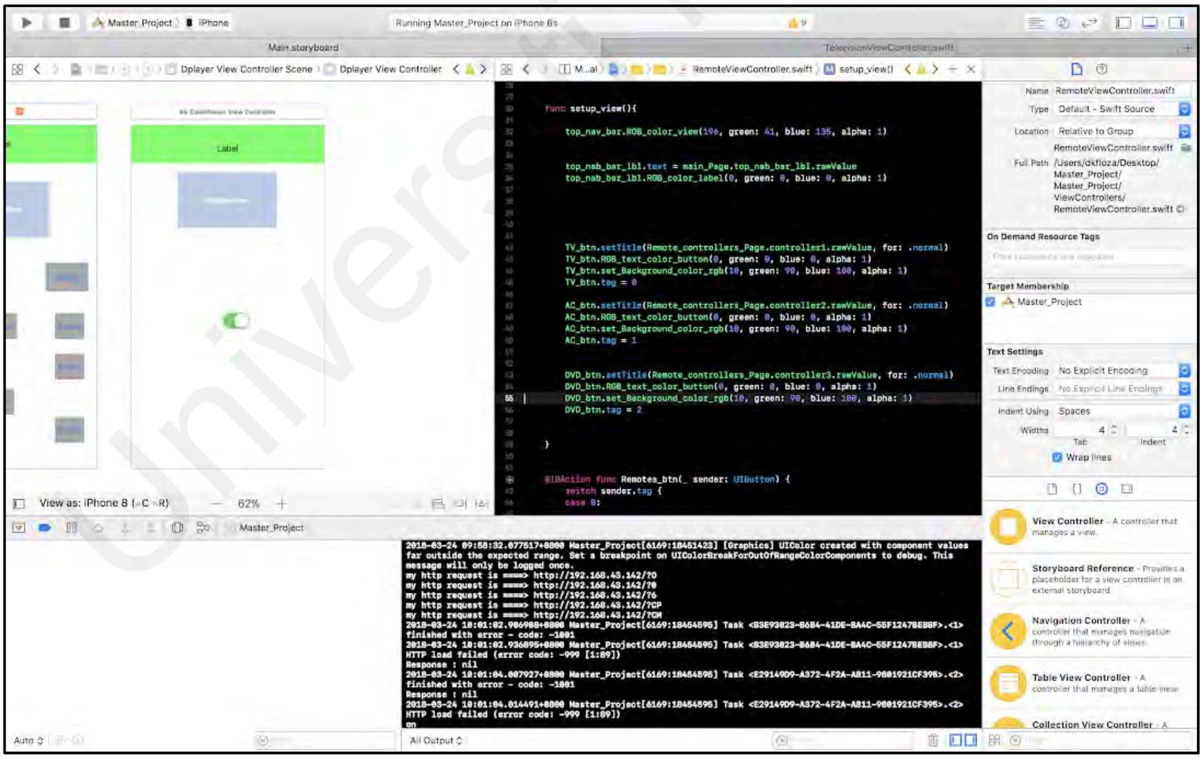
Universiti Malaya



```
//  Master_Project
//
//  Created by Danial Kosarifar on 30/01/2018.
//  Copyright © 2018 Danial Kosarifar. All rights reserved.
//

import UIKit




class TelevisionViewController: UIViewController , UICollectionViewDataSource, UICollectionViewDelegate , UICollectionViewDelegateFlowLayout
{

    @IBOutlet weak var rollback_btn: UIButton!
    let reuseIdentifier = "cell" // also enter this string as the cell identifier in the storyboard
    var items = [ images.channel_up_logo.rawValue ,
                  images.one_logo.rawValue,
                  images.two_logo.rawValue,
                  images.three_logo.rawValue,
                  images.channel_low_logo.rawValue,
                  images.four_logo.rawValue,
                  images.five_logo.rawValue,
                  images.six_logo.rawValue,
                  images.volume_up_logo.rawValue,
                  images.seven_logo.rawValue,
                  images.eight_logo.rawValue,
                  images.nine_logo.rawValue,
                  images.volume_low_logo.rawValue,
                  images.mute_logo.rawValue,
                  images.zero_logo.rawValue,
                  images.off_logo.rawValue
                   ]




    @IBOutlet weak var top_nav_bar: UIView!
    @IBOutlet weak var top_nab_bar_lbl: UILabel!
    @IBOutlet weak var tv_image: UIImageView!



    override func viewDidLoad() {
        super.viewDidLoad()
        setup_view()



    }



    func setup_view(){

        top_nav_bar.RGB_color_view( 196, green: 41, blue: 136, alpha: 1)


        top_nab_bar_lbl.text = television_controller_page.title.rawValue
        top_nab_bar_lbl.RGB_color_label(0, green: 0, blue: 0, alpha: 1)


        tv_image.image_mode_scaleAspectFit(images.tv_logo.rawValue)
        rollback_btn.background_img(images.rollback_logo.rawValue)


    }
```

Figure 5.17 : Xcode IDE with development screen for television remote control

Universiti Ma



```
    func setup_view(){

        top_nav_bar.RGB_color_view(196, green: 41, blue: 135, alpha: 1)


        top_nab_bar_lbl.text = main_Page.top_nab_bar_lbl.rawValue
        top_nab_bar_lbl.RGB_color_label(0, green: 0, blue: 0, alpha: 1)




        TV_btn.setTitle(Remote_controllers_Page.controller1.rawValue, for: .normal)
        TV_btn.RGB_text_color_button(0, green: 0, blue: 0, alpha: 1)
        TV_btn.set_Background_color_rgb(10, green: 90, blue: 100, alpha: 1)
        TV_btn.tag = 0

        AC_btn.setTitle(Remote_controllers_Page.controller2.rawValue, for: .normal)
        AC_btn.RGB_text_color_button(0, green: 0, blue: 0, alpha: 1)
        AC_btn.set_Background_color_rgb(10, green: 90, blue: 100, alpha: 1)
        AC_btn.tag = 1


        DVD_btn.setTitle(Remote_controllers_Page.controller3.rawValue, for: .normal)
        DVD_btn.RGB_text_color_button(0, green: 0, blue: 0, alpha: 1)
        DVD_btn.set_Background_color_rgb(10, green: 90, blue: 100, alpha: 1)
        DVD_btn.tag = 2



    }


    @IBAction func Remotes_btn(_ sender: UIButton) {
        switch sender.tag {
        case 0:
```

```
2018-03-24 09:58:32.077517+0800 Master_Project[6169:18451423] [Graphics] UIColor created with component values
far outside the expected range. Set a breakpoint on UIColorBreakForOutOfRangeColorComponents to debug. This
message will only be logged once.
my http request is ====> http://192.168.43.142/?O
my http request is ====> http://192.168.43.142/?8
my http request is ====> http://192.168.43.142/?6
my http request is ====> http://192.168.43.142/?CP
my http request is ====> http://192.168.43.142/?CM
2018-03-24 10:01:02.906988+0800 Master_Project[6169:18454595] Task <83E93023-B6B4-41DE-BA4C-55F1247BEB8F>.<1>
finished with error - code: -1001
2018-03-24 10:01:02.936895+0800 Master_Project[6169:18454595] Task <83E93023-B6B4-41DE-BA4C-55F1247BEB8F>.<1>
HTTP load failed (error code: -999 [1:89])
Response : nil
2018-03-24 10:01:04.007927+0800 Master_Project[6169:18454598] Task <E29149D9-A372-4F2A-AB11-9801921CF395>.<2>
finished with error - code: -1001
Response : nil
2018-03-24 10:01:04.014491+0800 Master_Project[6169:18454595] Task <E29149D9-A372-4F2A-AB11-9801921CF395>.<2>
HTTP load failed (error code: -999 [1:89])
on
```

Figure 5.18 : Xcode IDE with development screen for air conditioner remote control

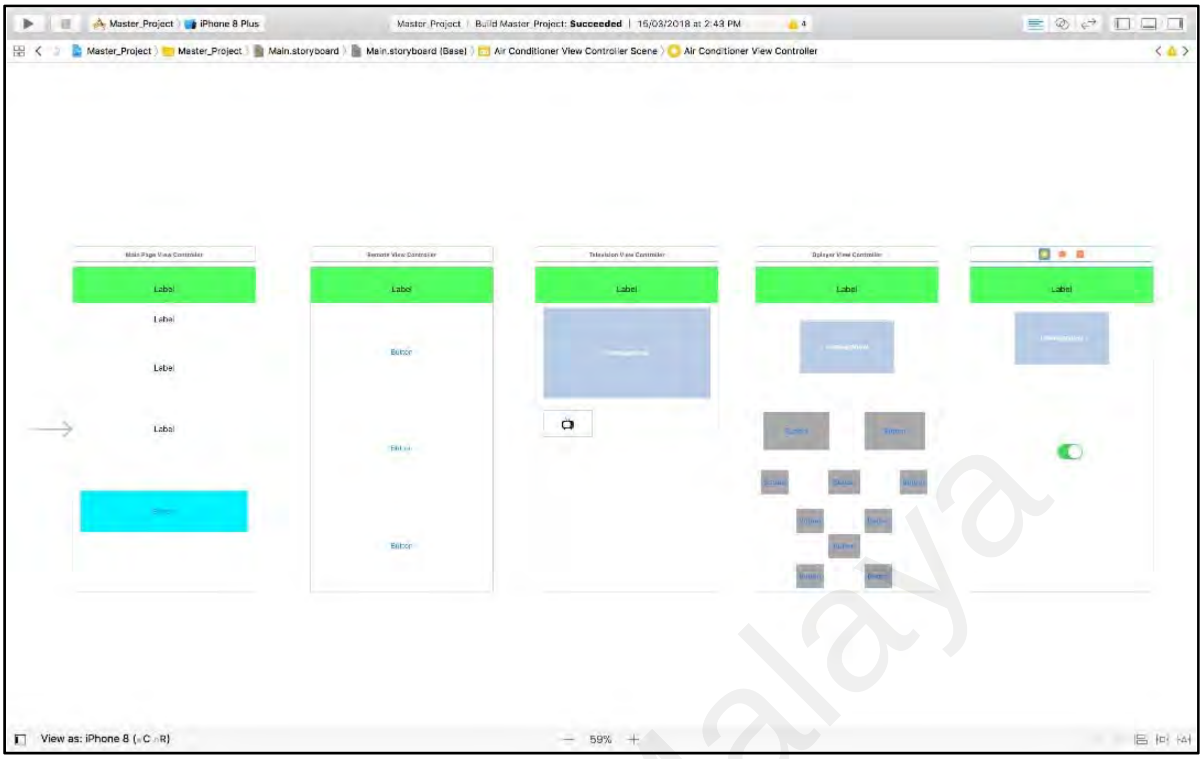
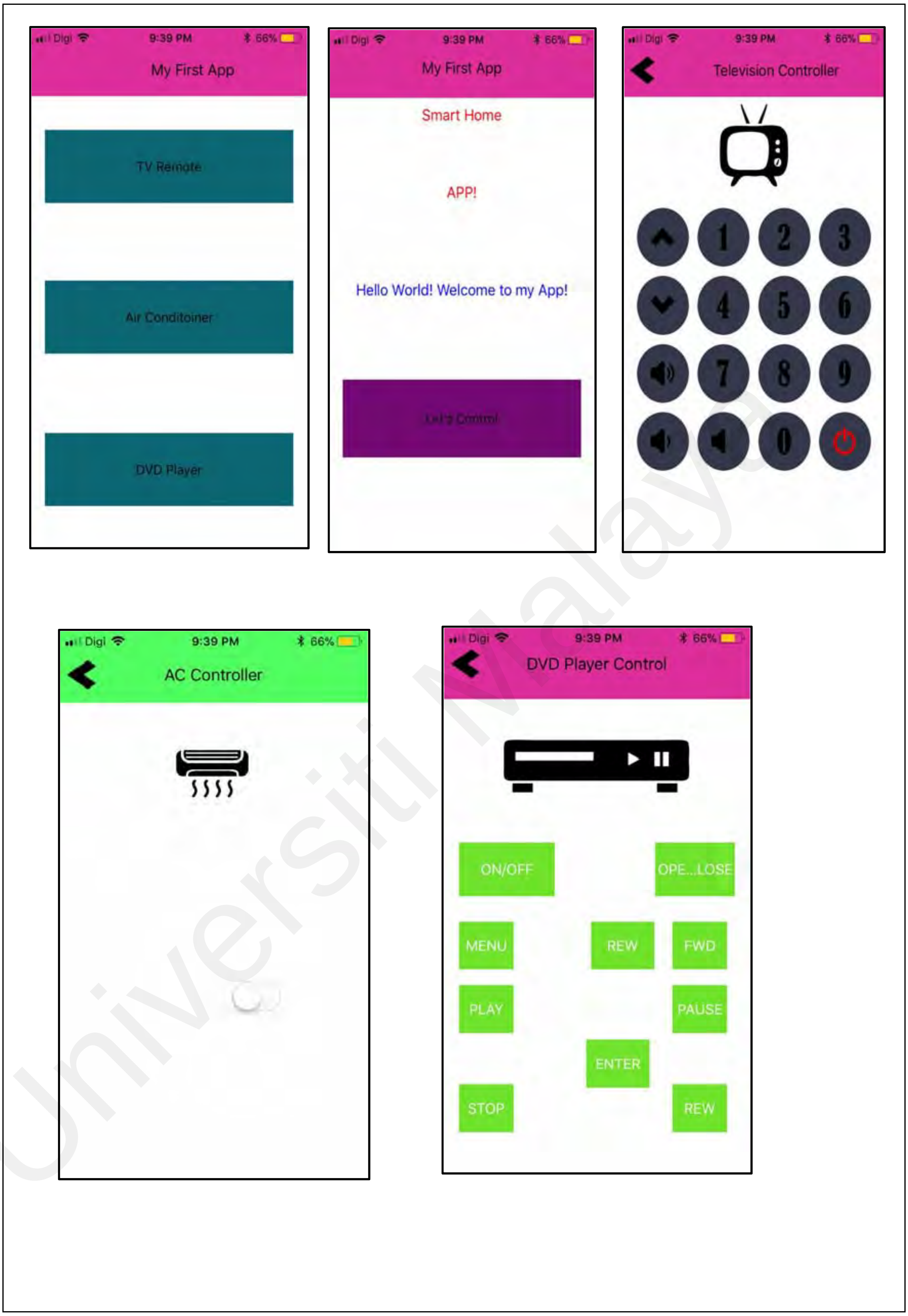

Master_Project
iPhone 8 Plus
Master_Project | Build Master_Project: Succeeded | 15/03/2018 at 2:43 PM
4
Master_Project
Master_Project
Main.storyboard
Main.storyboard (Base)
Air Conditioner View Controller Scene
Air Conditioner View Controller
Main Page View Controller
Remote View Controller
Television View Controller
Dplayer View Controller
Label
Label
Label
Label
Button
Button
Button
View as: iPhone 8 (wC hR)
59%

Figure 5.19 : Xcode IDE with user interface screen for the iOS mobile application

Universiti Malaya



Universiti Malaya

Figure 5.20: User Interface of iOS mobile application for IoT home application

The Best practices recommended for this phase were adapted which are:

- Best practices 1: The user matters – The user of the IoT home application is consumers.
- Best practices 2: Design the application: Designed the components and functions of this mobile application.
- Best practices 3: The interface – Designed the user interface. Avoided using too many graphics and animation. Assured the size of the buttons and images is not too large.
- Best practices 4: The performance – Tested the performance of the mobile application and ensured it to be within the design performance. The mobile application performance was monitored regularly.
- Best practices 5: Testing – The testing was done throughout the development of mobile application. The mobile application was tested with multiple users.
- Best practices 6: Connectivity and networks – Tested network and connection issues throughout the development and testing process.

Universiti Malaya

5.7 Chapter Summary

This chapter has discussed about the development process for IoT home application by applying the proposed guideline-based development approach. At each stage of the development process of IoT home application, the proposed guidelines in the form of best practices, checklist, sequence of steps and patterns have assisted in the development process. The development process went through few stages which are requirement analysis and review stage, development environment preparation stage, hardware setup stage, Arduino board programming stage, and mobile application development stage. The next chapter will discuss the evaluation of the IoT home application.

Universiti Malaya

CHAPTER 6: EVALUATION AND RESULTS

6.1 Introduction

The aim of this research is to propose guidelines to assist software developers in IoT Home application development process. Evaluation is a process of inquiring a product with the intention to evaluate it. Inquiring allows the tester to test the product, where the result is obtained based on the product behavior in reaction to the questioning of the tester (Regulwar & Gulhane, 2010). Validation of the outcome is part of the development process in any activity involving problem-solving (Adrion et al. 1982). Hence, questionnaire is a suitable evaluation method to evaluate the proposed guideline-based development approach and the developed IoT Home application.

This chapter describes the details of evaluating the proposed guideline-based development approach and the developed IoT Home application based on the proposed guidelines. This evaluation process was carried out with three means: 1) Expert evaluation and 2) User evaluation. The proposed guidelines were evaluated on its validity and effectiveness using a questionnaire approach. Meanwhile, the developed IoT Home application was also evaluated using the questionnaire approach but on its usability. The following sections present the details and the result of the evaluation.

Universiti Malaya

6.2 Evaluation

The evaluation of the proposed guideline-based development approach was performed in two ways: expert evaluation, and user evaluation. Furthermore, the evaluation of the usability of the developed IoT Home application was accomplished in two means: 1) expert evaluation, and 2) user evaluation.

The evaluation process from presenting the guideline-based development approach and the demonstrating the developed IoT Home application took around 20 – 30 minutes, followed by respondents answering the questionnaire form which took additional 2-3 minutes.

6.2.1 Expert Evaluation

The proposed approach was evaluated in terms of effectiveness and validity by five experts using the questionnaire method. The proposed approach together with the developed IoT Home application was presented to these experts. Subsequently, there were asked to answer the questions provided in an expert evaluation form, as in Appendix A.1. The expert evaluation form is divided into two parts. The first part illustrates the proposed approach and its effectiveness in assisting and directing software developers in developing IoT Home application. The second part of the expert evaluation form allows the experts to suggest any improvement and to comment on any problem identified in the proposed guidelines. The outcome of this expert evaluation is presented and further discussed in Section 6.3.

The five experts are comprised of the following segments:

1. Two experts who are currently studying Ph.D. in University Malaya.
2. Three experts who are currently working on IoT projects and had completed a few other IoT projects.

Universiti Malaya

6.2.2 User Evaluation

The user evaluation was participated by twenty users from different backgrounds and working fields. The developed IoT Home application system's usability was evaluated. The proposed approach was also evaluated by users but only those who have the knowledge of information technology or have been involved in software development. The questionnaire approach was used for user evaluation. The users were asked to answer the questions in the user evaluation form as in Appendix A.2. The user evaluation form is separated into four parts.

The first part comprises questions on the user's personal information: level of education and working field. The second part consists of questions on the effectiveness, affordance of the proposed approach and understandability on proposed approach. The third part illustrates the developed IoT Home application's ease-of-use and usability. The final part allows the user to comment on any problems experienced while using the IoT Home application, and on suggestion for improvement. The results of this user evaluation are detailed and discussed in Section 6.3.

The twenty users are comprised of the following segments:

1. Five users who are students currently studying Master of Computer Science and Ph.D. of Computer Science in University Malaya.
2. Eight users who are software developers working in the different types of software development.
3. Two users who are housewife.
4. Five IoT experts who have evaluated the proposed guideline-based development approach.

Universiti Malaya

6.3 Results and Discussion

The results of expert and user evaluations are presented and discussed in the following subsections. The list of the evaluation result for the both expert and user evaluation is presented in Appendix A.3.

6.3.1 Expert Evaluation result and discussion

Five experts have evaluated the effectiveness and validity of the proposed approach using the questionnaire. The proposed approach and the developed IoT Home application were presented to the five experts.

Table 6.1 : Questions and responses by expert evaluation on the proposed approach

Part 1:	Evaluation of the Guideline-based development approach			
No.	Questions	Number of experts		
		Yes	No	Not Sure
1.	Would this guideline-based development approach help to speed up the IoT Home Automation development process?	5		
2.	Would this guideline-based approach guide software developers in the phases of the IoT home application development process?	5		
3.	Do you think a junior IoT software developer or a developer who is not familiar with the setting up of the required hardware device for IoT home application development will be able to do it by themselves without depending on hardware developers or by referring to the proposed guidelines?	3		2
4.	Would the approach help the software developers in choosing a suitable programming language and IDE tools for the Android and IOS mobile application development?	5		
5.	Do you think that the source code for the Arduino Board and Android & IOS mobile application is reusable?	4		1
6.	Do you think that the proposed guideline-based development approach has enough details or information required to assist the software	5		

Universiti Malaya

	developers on the IoT home application development without a need to look for other resources for development guidance?			
7.	Does the research accomplish the objectives by proposing this guideline-based development approach?	5		

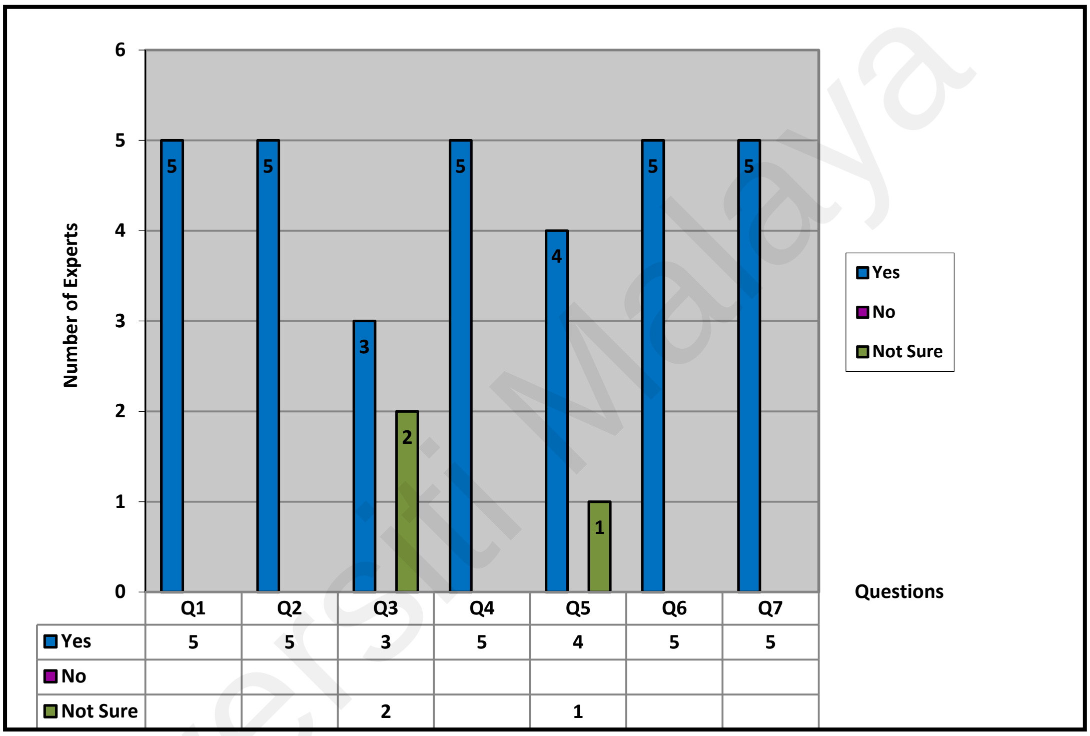


Figure 6.1 : Response frequencies by experts for proposed approach

Universiti Malaya

Table 6.1 shows the responses from experts to the questions on the proposed approach. Figure 6.1 presents the frequencies of responses given in Table 6.1 for the questions asked in the expert evaluation form. It clearly shows from Figure 6.1 that most of the experts respond “Yes”, only three experts response “Not Sure” for two questions, and no experts respond “No” to any questions. The same person has responded “Not Sure” for two questions. This shows that two experts (represent 40% of the five surveyed experts) are not sure about two questions (represent 28% of the questions). Therefore, around 8% of the total answers were “Not Sure” compared to 92% were “Yes”. Overall, most of the experts agree that the proposed approach is valid and effective.

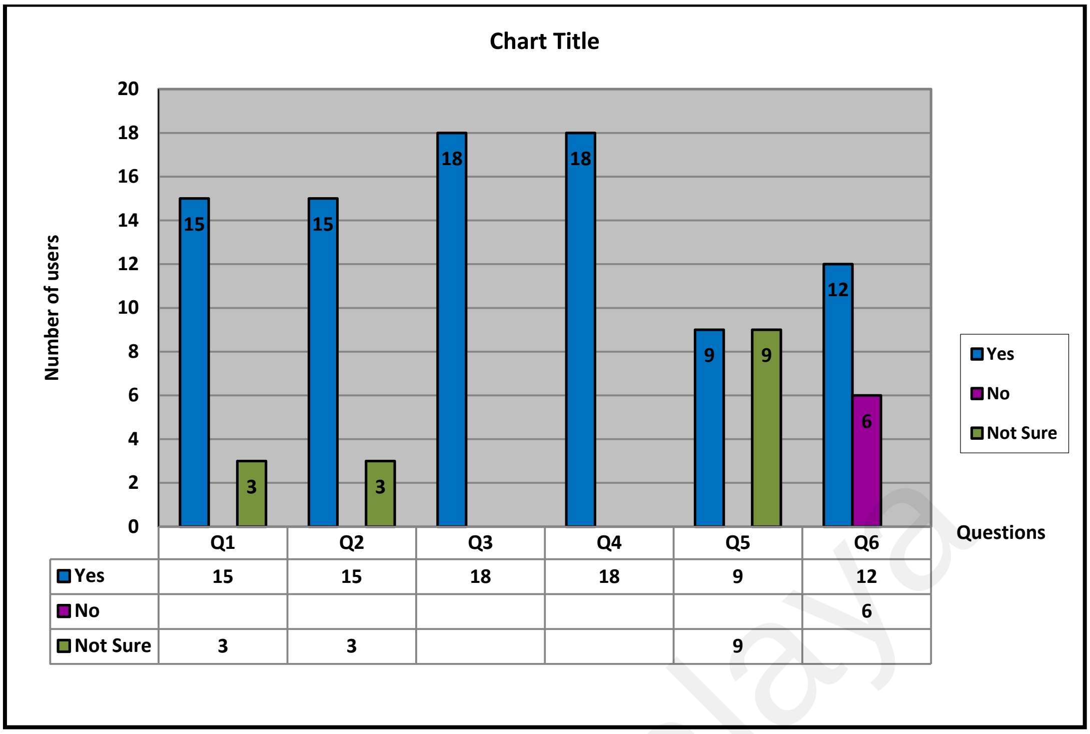
6.3.2 User Evaluation results and discussion

Twenty users evaluated the developed IoT Home application system’s usability and eighteen users have evaluated the proposed approach. The questionnaire approach was also used for user evaluation.

Table 6.2 : Questions and responses by user evaluation on the proposed guideline-based development approach

Part 2:	**Guideline-based development approach**			
No.	Questions	Number of users		
		Yes	No	Not Sure
1.	Do you find it easy to understand the approach that was proposed?	15		3
2.	Do you find the guidelines consistent?	15		3
3.	Will you use the guideline-based approach as guidance when developing an IoT Home application system?	18		
4.	Do you find the approach complete and suitable to be referred throughout the development process?	18		
5.	Do you think that the source code for the Arduino Board and Android & IOS mobile application is reusable?	9		9
6.	Do you think with this guideline-based development approach, it is not required to find other resources as guidance or references?	12	6	

Universiti Malaya



	Q1	Q2	Q3	Q4	Q5	Q6
🟦 Yes	15	15	18	18	9	12
🟪 No						6
🟩 Not Sure	3	3			9	

Figure 6.2 : Response frequencies by users for proposed guideline-based development approach

Universiti Malaya

Table 6.2 shows the responses from the users to the questions on the proposed approach. Figure 6.2 displays the response frequencies as given in Table 6.2 on the questions in the user evaluation form. Figure 6.2 clearly illustrates that most of the users respond “Yes”, nine users respond “Not Sure” to three questions, and six users respond “No” to one question. Three users answered “No” and “Not Sure” to four questions. This indicates that the three users are not familiar with IoT Home application development. The six users who answered “No” to the question No. 6 are not confident in depending on the proposed approach for the IoT Home application development. Figure 6.3 displays the percentages of answers for user evaluation on the proposed guideline-based development approach. The figure indicates that 5% of the users answered “No” and 14% were “Not Sure”, meanwhile 81% (majority) of the answers were “Yes”. Overall, the evaluation from the users indicates that most of the users agree that the proposed approach is dependable and reliable to apply for the development of IoT Home application.

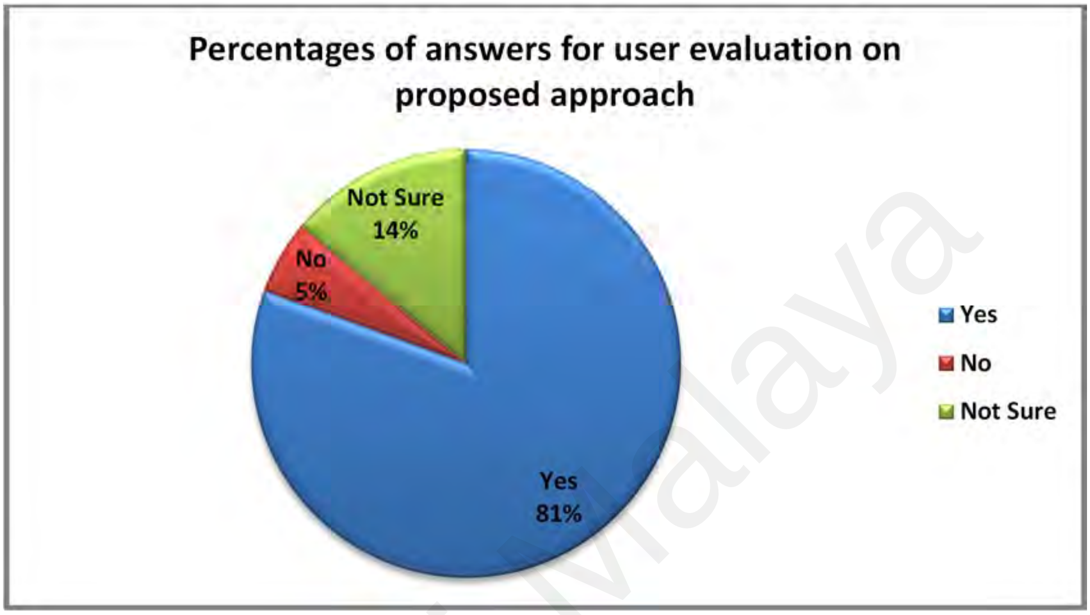


Figure 6.3 : Percentages of answers by users for proposed guideline-based development approach

Universiti Malaya

Table 6.3 : Questions and responses by user evaluation on the developed system

Part 3:	**IoT home application system**			
No.	Questions	Number of users		
		Yes	No	Not Sure
1.	Is the application system easy to use?	20		
2.	Is the interface of the application system pleasant?	20		
3.	Is the navigation of the application frictionless?	20		
4.	Does the system have all functions required?	19		1
5.	Is the system not too complex to understand and use?	20		
6.	Do the system and the device respond with a reasonable time while using any of the functions?	20		
7.	Integration: does the system application connect to the microcontroller and to the device?	20		
8.	Does the system load within a reasonable time?	18	1	1
9.	Is the system user-friendly?	20		

10.	Overall, are you satisfied with the system?			20		
11.	How fast were the response of the system and the devices while performing the function of the system?	Very Fast 3	Fast 17	Average	Slow	Very Slow

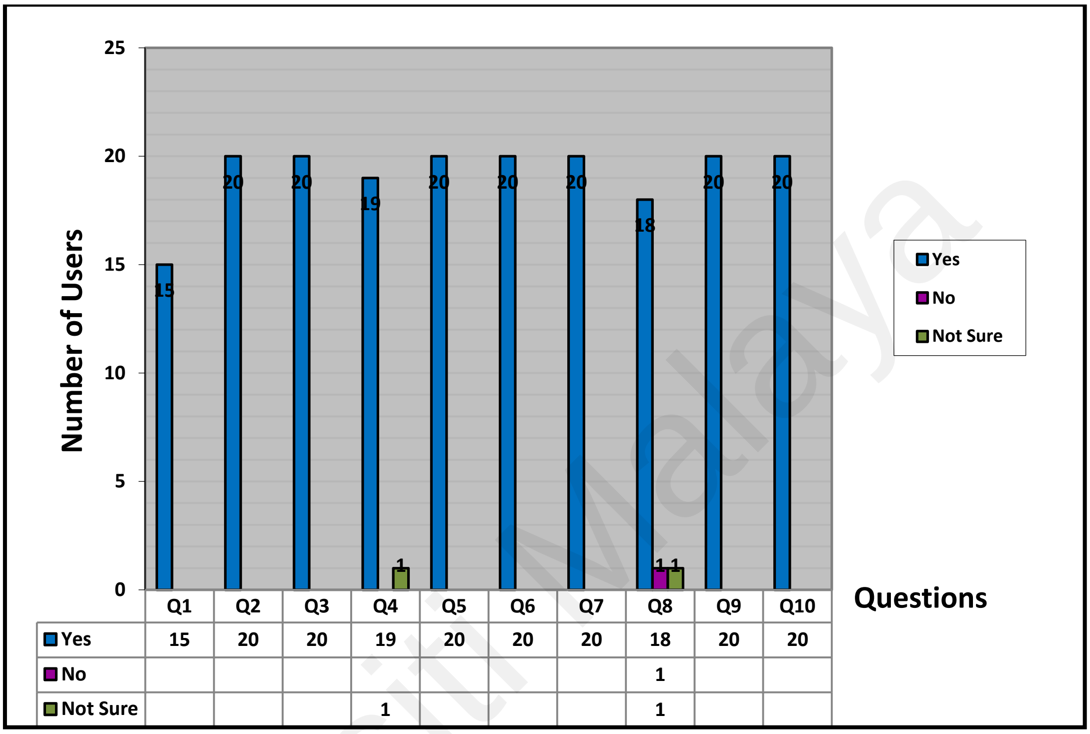


Figure 6.4 : Response frequencies by users for developed IoT Home application

Universiti Malaya

Table 6.3 shows the responses from users to the developed IoT Home applications questionnaire approach in terms of usability. Figure 6.4 displays the response frequencies given in Table 6.3 for questions in the user evaluation form. It shows from Figure 6.4 where all users respond “Yes” excluding two user responds “Not Sure” to two questions. Meanwhile, one of the users responds “No”. The response to the function of the developed application is evaluated as “Very fast” by three users and “Fast” by seventeen users. Overall, the result illustrates the developed IoT Home application satisfied the users with the available functions and features.

6.4 Chapter Summary

This chapter has presented the evaluation process of the proposed guideline-based development approach and the developed IoT Home application. The evaluation method was accomplished via two activities: 1) expert evaluation, and 2) user evaluation. The outcome from the expert evaluation illustrates that experts endorse the proposed guideline-based development approach as a valid and effective tool. The expert evaluation and user evaluation results illustrate that the developed system by implementing the proposed approach is useful and usable.

Meanwhile, user evaluation indicates the proposed guidelines are understandable and suitable to be used as guidance on developing IoT Home application systems.

As an outcome, the proposed guideline-based development approach helps in assisting developers on developing IoT Home application from requirement review stage till the mobile application development stage.

Universiti Malaya

CHAPTER 7: CONCLUSION

7.1 Strengths and contribution

The previous works focus on providing framework, programming technique, and best practices for IoT Home application development. However, there is lack of guidelines in setting up the hardware for IoT Home application and mobile application development for IoT. This research has proposed a guideline-based development approach for setting up the IoT hardware without depending upon an electronic engineer for setting hardware equipment for IoT Home application which includes the setup of Arduino microcontroller, sensors, as well as the guidelines for Arduino Programming.

Besides that, most of the previous works concentrated on the programming stage; the guidelines in previous work did not cover or provide guidelines for every stage of IoT development. This research has proposed a guideline-based development approach that has satisfied the above need by providing guidelines for every stage of IoT Home application, which are: 1) Requirements review stage, 2) Development environment preparation stage, 3) IoT Hardware setup stage, 4) Arduino board programming stage, and 5) Mobile application development stage. Apart from that, all the guidelines for every stage are stored in one place so that it is easy to be referred by the software developers throughout the development process. Specifically, it spares software developers from investing their time in looking for scattered guidelines from numerous sources.

Choosing a programming language can be a major barrier, for those who are lack of necessary information to make a right choice. The choice made by the software developers can break the project or make the application successful. There are various programming languages to decide from, and one shouldn’t consider single aspect when choosing the programming language. Each programming language has different features

Universiti Malaya

that should be well thought-out. As such, this research has proposed an approach that guides the software developers in deciding on an appropriate programming language in the perspective of Android and iOS's portable criteria for IoT Home application development.

7.2 Limitations

This research has perceived potential limitation in the proposed guideline-based development approach that can be improved in future:

- There are many choices of microcontroller development board for IoT that are available at current market including Arduino, Raspberry Pi, and AdaFruit Flora. Arduino and Raspberry Pi are known to be the most favorite and popular platform among IoT developers. Arduino is least difficult yet the most intense prototyping platform and is really more qualified as an interface for sensors. Meanwhile, Raspberry Pi 3 is a powerful computing platform and certainly the most reasonably priced. In this study, the proposed guideline for IoT hardware setup and board programming has concentrated only on the microcontroller which is Arduino.
- The number of expert participants involved in evaluating the proposed guideline-based development approach is limited to five experts.

7.3 Future Work

The proposed guideline-based development approach has accomplished its objectives; however there is still room for improvement in the proposed approach based on the limitation identified in the earlier section, as stated below:

Universiti Malaya

- There are other microcontroller boards other than Arduino that are ideal to be used for IoT development such as: 1) Raspberry Pi, and 2) AdaFruit Flora. As currently, the proposed approach has concentrated on Arduino, software developers who are using this approach have limited choice or no other choice to just use the Arduino as microcontroller for their project. As such, there is a need in future to add guidelines on setting up IoT hardware which involve other microcontrollers. By proposing guidelines and including it in the approach on other microcontrollers, it will provide more choices for software developers that can be used for their project. The pros and cons of each microcontroller are also to be included in this approach as it will assist them to choose the most suitable microcontroller for their projects. Overall, a guideline for setting up IoT hardware that involves other microcontrollers as well as the best practices on choosing the most suitable microcontroller is to be considered as a future work.
- The expert evaluation should be conducted with more than ten expert participants. To have confidence that the results of the questionnaire are representative; large number of participants is important in an evaluation process. Despite that, the larger size of participants gives greater power for the questionnaire evaluation and the accuracy of the results will be improved.

Universiti Malaya

7.4 Conclusion

The IoT has started to play an important role in current time and future. IoT is considered having the power to change the world as major business systems will integrate with IoT. Due to this, IoT development in various areas such as, health, home environment, industrial and etc have become a trend in the currently. An IoT development is considered as new or a different type of development for software developers compared to the web development and mobile application development as it involves both hardware and software development.

Thus, junior software developers or season software developers who are lack of expertise on electronic development will or might have difficulties in developing an IoT Home application development. They will encounter difficulties in gathering the requirements related to IoT Home application projects due to the lack of experience or exposure to IoT development. Other than that, the setting of development environment and setting up of IoT hardware equipments such as microcontrollers and sensors will be a stage where the software developers most certainly will face problems on attending to it.

Besides, programming involves a microcontroller as well as choosing a suitable programming language for IoT mobile applications is likely to take time for inexperienced software developers. Therefore, this research provides a solution to the problems indicated above and guides software developers in IoT Home application development process. This research has proposed a guideline-based development approach that provides guidelines for six stages of IoT Home application development process. The proposed approach involves using suitable and task-supported forms such as: 1) best practices, 2) sequence order steps, 3) patterns, and 4) checklist.

An expert and user evaluation method is used to evaluate the proposed guidelines and the developed system. The preparatory outcomes demonstrated that the proposed guideline-based development approach is helpful and successful in directing the software developers all through the development process of IoT Home application, and the developed IoT Home application by applying the proposed guidelines is helpful in controlling the home appliances.

Universiti Malaya

Overall, this dissertation contributes to the IoT Home application development. This research fills a gap on IoT Home application developing area by proposing guideline-based development approach to assist and provide direction while developing IoT Home application system.

Universiti Malaya

REFERENCES

Abivandhana, M., Divya, K., Gayathri, D., & RuhinKouser, R. (2017). Smart Home Automation Based on IOT and Android Technology. *International Journal of Engineering Science*, *5943*

Adrion, W.R., Branstad, M.A. &Cherniavsky, J.C., 1982. Validation, verification, and testing of computer software. *ACM Computing Surveys (CSUR)*, 14(2), 159–192.

Alur, R., Berger, E., Drobnis, A. W., Fix, L., Fu, K., Hager, G. D., ... & Rexford, J. (2016). Systems computing challenges in the Internet of Things. *arXiv preprint arXiv:1604.02980*.

Atzori, L., Iera, A., & Morabito, G. (2010). The internet of things: A survey. *Computer networks*, *54*(15), 2787-2805.

Beal, J., Pianini, D., & Viroli, M. (2015). Aggregate programming for the internet of things. *IEEE Comput*, *48*(9), 22-30.

Best practices for IoT development Applying lessons from mobile development to your IoT projects Andrew Fisher September 02, 2015

Charpenay, V., Käbisch, S., Anicic, D., & Kosch, H. (2015, October). An ontology design pattern for iot device tagging systems. In *Internet of Things (IOT), 2015 5th International Conference on the* (pp. 138-145). IEEE.

Charumathi, K., Shareen, H. H., Sushmitha, R., Suriya, R., & Prasanna, K. (2018). CONTROLLING AND MONITORING OF SMART HOME APPLIANCES USING IoT. *identity*, *2*(3).

Consel, C., & Kabac, M. (2014). Internet of Things: A Challenge for Software Engineering. *ERCIM News, Smart Cities,*, (98 (2014)), 20-21.

Da Xu, L., He, W., & Li, S. (2014). Internet of things in industries: A survey. *IEEE Transactions on Industrial Informatics*, *10*(4), 2233-2243.

Díaz, M., Garrido, D., & Reyna, A. (2009). One step closer to the Internet of things: SMEPP. In *Future Internet Symposium FIS*.

Universiti Malaya

Ebert, C., & De Neve, P. (2001). Surviving global software development. *IEEE software*, *18*(2), 62-69.

Eclipse IOT Website (2015) Available online at: 651 http://iot.eclipse.org

Farrell, A. (2007). Selecting a Software Development Methodology based on Organizational Characteristics. *An Essay Submitted in Partial Fulfillment of the Requirements for the Degree of "Master of Science in Information Systems", Athabasca University, Athabasca*.

Fisher, A. (2 September, 2015). Best practices for IoT development. pp. 1-6 --- https://www.ibm.com/developerworks/library/iot-mobile-practices-iot-success/iot-mobile-practices-iot-success-pdf.pdf

Frenzel, L., Wong, W., Attractive, M., Mannion, P., & Guerra, M. (2016). Comparing Ada and C.

Fortino, G., Guerrieri, A., Russo, W., & Savaglio, C. (2015, October). Towards a development methodology for smart object-oriented IoT systems: a metamodel approach. In *Systems, Man, and Cybernetics (SMC), 2015 IEEE International Conference on* (pp. 1297-1302). IEEE.

Ghosh, K., Kalbhor, R., Tejpal, D., & Haral, S. (2015). WIRELESS HOME AUTOMATION TECHNOLOGY (WHAT) USING INTERNET OF THINGS (IOT). *International Journal of Technical Research and Applicationse-ISSN*, 2320-8163.

Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, *29*(7), 1645-1660.

Gyrard, A., Bonnet, C., Boudaoud, K., & Serrano, M. (2015, December). Assisting IoT Projects and Developers in Designing Interoperable Semantic Web of Things Applications. In *2015 IEEE International Conference on Data Science and Data Intensive Systems* (pp. 659-666). IEEE.

Gyrard, A., Serrano, M., & Atemezing, G. A. (2015, December). Semantic web methodologies, best practices and ontology engineering applied to Internet of Things. In *Internet of Things (WF-IoT), 2015 IEEE 2nd World Forum on* (pp. 412-417). IEEE.

Universiti Malaya

Heer, J., & Agrawala, M. (2006). Software design patterns for information visualization. *IEEE transactions on visualization and computer graphics*, *12*(5), 853-860.

Hiesgen, R. (2015). Embedded Actors–A Better Abstraction for Distributed Messaging in the IoT Project.

Hneif, M., & Ow, S. H. (2009). Review of agile methodologies in software development. *International Journal of Research and Reviews in Applied Sciences*, *1*(1), p1-8.

IoT Developer Survey, April 2016 available at - http://iot.ieee.org/images/files/pdf/iot-developer-survey-2016-reportfinal.pdf

Ismaeel, A. G., & Kamal, M. Q. (2017, April). Worldwide auto-mobi: Arduino IoT home automation system for IR devices. In *Current Research in Computer Science and Information Technology (ICCIT), 2017 International Conference on* (pp. 52-57). IEEE.

Kess, P., & Kropsu-Vehkaperä, H. (2016). Standardization with Iot (Internet-of-Things). In *Managing Innovation and Diversity in Knowledge Society Through Turbulent Time: Proceedings of the MakeLearn and TIIM Joint International Conference 2016* (pp. 1069-1076). ToKnowPress.

Khan, R., Khan, S. U., Zaheer, R., & Khan, S. (2012, December). Future internet: the internet of things architecture, possible applications and key challenges. In *Frontiers of Information Technology (FIT), 2012 10th International Conference on* (pp. 257-260). IEEE.

Klimeš, C., & Procházka, J. (2006). New Approaches in Software Development. *Acta Electrotechnica et Informatica No*, *6*(1), 2.

Kortuem, G., Bandara, A. K., Smith, N., Richards, M., & Petre, M. (2013). Educating the Internet-of-Things generation. *Computer*, *46*(2), 53-61.

Lee, I., & Lee, K. (2015). The Internet of Things (IoT): Applications, investments, and challenges for enterprises. *Business Horizons*, *58*(4), 431-440.

Li, B., & Yu, J. (2011). Research and application on the smart home based on component technologies and Internet of Things. *Procedia Engineering*, *15*, 2087-2092.

Universiti Malaya

Lü, J., Rosenblum, D. S., Bultan, T., Issarny, V., Dustdar, S., Storey, M. A., & Zhang, D. (2015). Roundtable: The Future of Software Engineering for Internet Computing. *IEEE Software*, *32*(1), 91-97.

Malik, H., & Hneif, M. (2010). *Guideline-based approach for improving the achievement of non-functional attributes of software* (Doctoral dissertation, University of Malaya).

Mazzei, D., Montelisciani, G., Baldi, G., & Fantoni, G. (2015, December). Changing the programming paradigm for the embedded in the iot domain. In *Internet of Things (WF-IoT), 2015 IEEE 2nd World Forum on* (pp. 239-244). IEEE.

Miorandi, D., Sicari, S., De Pellegrini, F., & Chlamtac, I. (2012). Internet of things: Vision, applications and research challenges. *Ad Hoc Networks*, *10*(7), 1497-1516.

Mzahm, A. M., Ahmad, M. S., Tang, A. Y., & Ahmad, A. (2015, August). Software analysis for Agents of Things (AoT) applications. In *Agents, Multi-Agent Systems and Robotics (ISAMSR), 2015 International Symposium on*(pp. 12-17). IEEE.

Namiot, D., & Sneps-Sneppe, M. (2014). On IoT Programming. *International Journal of Open Information Technologies*, *2*(10).

Oriwoh, E., Sant, P., & Epiphaniou, G. (2013). Guidelines for internet of things deployment approaches–the thing commandments. *Procedia Computer Science*, *21*, 122-131.

Patel, P., & Cassou, D. (2015). Enabling high-level application development for the Internet of Things. *Journal of Systems and Software*, *103*, 62-84.

Patel, P., Luo, T., & Bellur, U. (2016). Evaluating a Development Framework for Engineering Internet of Things Applications. *arXiv preprint arXiv:1606.02119*.

Patel, P., Pathak, A., Teixeira, T., & Issarny, V. (2011, December). Towards application development for the internet of things. In *Proceedings of the 8th Middleware Doctoral Symposium* (p. 5). ACM.

Piyare, R. (2013). Internet of things: ubiquitous home control and monitoring system using android based smart phone. *International Journal of Internet of Things*, *2*(1), 5-11.

Universiti Malaya

Piyare, R., & Lee, S. R. (2013). Smart home-control and monitoring system using smart phone. *ICCA 2013, ASTL*, *24*, 83-86.

Qanbari, S., Pezeshki, S., Raisi, R., Mahdizadeh, S., Rahimzadeh, R., Behinaein, N., ... & Yaghini, A. (2016, April). IoT Design Patterns: Computational Constructs to Design, Build and Engineer Edge Applications. In *Internet-of-Things Design and Implementation (IoTDI), 2016 IEEE First International Conference on* (pp. 277-282). IEEE.

Ramachandran, M. (2012). Guidelines based software engineering for developing software components. *Journal of Software Engineering and Applications*, *5*(01), 1.

Reddy, E. R., & Sankara, K. (2018). Internet of Things Based Home Automation Control System Using Raspberry Pi.

Regulwar, G. B., & Gulhane, V. S. (2010). Software Testing Practices. *International Journal of Computer Applications*, *1*(2), 1-7.

Reinfurt, L., Breitenbücher, U., Falkenthal, M., Leymann, F., & Riegg, A. (2016, July). Internet of things patterns. In *Proceedings of the 21st European Conference on Pattern Languages of Programs* (p. 5). ACM.

Ribeiro, A. D. R. L., de Almeida, F. M., Moreno, E. D., & Montesco, C. A. (2016, September). A management architectural pattern for adaptation system in Internet of Things. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2016 International* (pp. 576-581). IEEE.

Riehle, D., & Züllighoven, H. (1996). Understanding and using patterns in software development. *TAPOS*, *2*(1), 3-13.

Rouse, M. (2007, February). *Best Practice*. Retrieved from TechTarget: http://searchsoftwarequality.techtarget.com/definition/best-practice

Rouse, M. (2016, July). *Internet of Things (IoT)*. Retrieved from TechTarget: http://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT

Rouse, M. (2016, July). *Internet of Things (IoT)*. Retrieved from TechTarget: http://internetofthingsagenda.techtarget.com/definition/home-automation

Universiti Malaya

Singh, N., Bharti, S. S., Singh, R., & Singh, D. K. (2014, August). Remotely controlled home automation system. In *Advances in Engineering and Technology Research (ICAETR), 2014 International Conference on* (pp. 1-5). IEEE.

Sivieri, A., Mottola, L., & Cugola, G. (2012, June). Drop the phone and talk to the physical world: Programming the internet of things with Erlang. In*Proceedings of the Third International Workshop on Software Engineering for Sensor Network Applications* (pp. 8-14). IEEE Press.

Soundarajan, N., & Hallstrom, J. O. (2004, May). Responsibilities and rewards: Specifying design patterns. In *Proceedings of the 26th International Conference on Software Engineering* (pp. 666-675). IEEE Computer Society.

Sundmaeker, H., Guillemin, P., Friess, P., & Woelfflé, S. (2010). Vision and challenges for realising the Internet of Things. *Cluster of European Research Projects on the Internet of Things, European Commision*, *3*(3), 34-36.

Stojkoska, B. L. R., & Trivodaliev, K. V. (2017). A review of Internet of Things for smart home: Challenges and solutions. *Journal of Cleaner Production*, *140*, 1454-1464.

Vinay Sagar, K. N., & Kusuma, S. M. (2015). Home automation using internet of things. *Int. Res. J. Eng. Technol*, *2*, 1965-1970.

Vlissides, J., Helm, R., Johnson, R., & Gamma, E. (1995). Design patterns: Elements of reusable object-oriented software. *Reading: Addison-Wesley*, *49*(120), 11.

Wadhwani, S., Singh, U., Singh, P., & Dwivedi, S. (2018). Smart Home Automation and Security System using Arduino and IOT. *International Research Journal of Engineering and Technology*, *5*(02), 1357-1359.

Wasserman, A. I. (2010, November). Software engineering issues for mobile application development. In *Proceedings of the FSE/SDP workshop on Future of software engineering research* (pp. 397-400). ACM.

Whitmore, A., Agarwal, A., & Da Xu, L. (2015). The Internet of Things—A survey of topics and trends. *Information Systems Frontiers*, *17*(2), 261-274.

Wu, K., He, X., Chen, X., & Huang, G. (2017). Runtime Model Based Approach to Smart Home System Development. *arXiv preprint arXiv:1707.08771*.

Universiti Malaya

Zambonelli, F. (2016). Towards a general software engineering methodology for the Internet of Things. *arXiv preprint arXiv:1601.05569*.

Zambonelli, F. (2016). Towards a Discipline of IoT-Oriented Software Engineering. In *WOA* (pp. 1-7).

Universiti Malaya